The ECU+
Fuel/Timing/Boost/Datalogging
Piggyback Engine Computer

Installation/Users Manual

Manual Revision 2.16 - Dec 21, 2011

Table of Contents

1 Introduction

Thanks for purchasing the ECU+ fuel, timing and boost control computer. The ECU+ is an inexpensive, full-featured piggyback computer that modifies the MAS airflow and the cam and crank angle sensor signals on your 1990-1999 4G63-equipped turbo Eclipse, Talon, Laser or Galant VR-4, or 2003-2006 Lancer Evolution, giving you complete control over fuel delivery and timing of your vehicle. The ECU+ is typically used on high-performance vehicles with larger turbos, upgraded exhausts and higher-flow injectors to fine-tune your vehicle's performance.

The ECU+ can modify the fuel and timing of your engine at some 133 "points," depending on the engine's air flow, RPM and throttle position. In addition, the ECU+ supports full datalogging of most engine sensors at 25 samples per second as well as complete OBD-II functionality. Datalogged information can be viewed and analyzed with the included Windows software.

2 Warning

The ECU+ system is an sophisticated device intended for off-road use, and is not in any way certified for use on public streets. The ECU+ provides you with the ability to adjust the air/fuel and timing mixture of the engine in your car. If used improperly, the ECU+ **can destroy your engine**. Neither the dealer nor the manufacturer of the ECU+ may be held responsible for any damage to your vehicle as a result of the installation, use or misuse of this product. Additionally, you should never adjust your engine's settings or attempt to “run” the ECU+ Win software while the vehicle is in motion.

3 System Requirements

The ECU+ supports the following vehicles:

- 1990-1991 Mitsubishi Galant VR-4
- 1990-1994 Plymouth Laser, Mitsubishi Eclipse or Eagle Talon, AWD or FWD, turbo only
- 1995-1999 Mitsubishi Eclipse or Eagle Talon, AWD or FWD, turbo only
- 2003-2005 Mitsubishi Lancer Evolution VIII
- 2006 Mitsubishi Lancer Evolution IX

The ECU+ head unit connects via a serial port to a laptop PC for data display, logging and analysis.

The ECU+ Win software requires a Intel, AMD or compatible PC running Windows 98, 98SE, ME, 2000, XP or Vista. (At the time of this writing, Windows 7 is not yet released, but should work fine as it's almost identical to Vista. Report any Windows 7 issues to support@ecuplus.com.) A Pentium II/366 Mhz or faster CPU is recommended. If the PC is a laptop, it can be used directly with the ECU+ head unit via a serial connection (be that a true serial port, or a pseudo-serial port, like a USB-to-serial adapter).

4 Packing List

When you purchased the ECU+, you should have received the following items:

- ECU+ head unit – this is the “silver and black box.” When shipped in the plug-n-play (PnP) version, the head unit is pre-wired into a adapter harness.

- A software CD-ROM containing the ECU+ Windows software, as well as this manual and other documentation.
- Two wiring harnesses for connection to the car – one for high-current I/O and one for low-current I/O. For the plug-n-play (PnP) version of the ECU+, these will be pre-wired into a PnP harness.
- Wire ties.
- A 9-pin straight-through serial cable. Male to female. This is used to interface between the ECU+ head unit and the laptop.
- A null modem adapter, female to female. This will adapt a Palm organizer to the serial cable when a future release of the ECU+ Palm software becomes available.

5 Head Unit Hardware Installation

The ECU+ "head unit" is the black and silver box of the ECU+ system. This device connects into your car's wiring harness and intercepts various engine sensor signals. The head unit modifies theses signals based on your inputs, and passes them on to the stock ECU. This section will explain how to install the ECU+ head unit in your car.

5.1 Tools Required

The ECU+ head unit can be installed by anyone with some basic electrical experience. If you can hook up a new car stereo, the ECU+ head unit installation will be a breeze. To complete the installation, you'll need the following tools:

- A set of common screwdrivers and a socket set (for removing the stock ECU).
- Wire cutters and a small knife.
- Electrical tape.
- A pencil-type soldering iron and solder.
- A digital multimeter for measuring voltages.
- Lots of patience.

5.2 Wiring Basics

To install the non-PnP ECU+, you'll be splicing into the stock wiring harness that connects the engine sensors to the stock ECU. You'll make one of two types of connections: tapping or splicing into the stock wiring harness.

5.2.1 Tapping Wires

For most of the signals that the ECU+ head unit needs to see, you'll be strictly tapping into the stock wiring harness. That is, you won't be cutting the wires at all, but just stripping back enough insulation so that you can "tap" the wire. Here's what you'll do:

1. Strip about an inch of insulation from the stock ECU wire.
2. Strip an inch and a half of insulation from the end of the corresponding ECU+ wire.
3. Twist the bare ECU+ wire around the bare stock wire, then solder the two together to make a solid connection.

4. Wrap this exposed connection with electrical tape.

5.2.2 Splicing Wires

A few of the ECU+ head unit connections require that you actually splice into the stock wiring harness. This involves cutting a wire in the stock harness and connecting a ECU+ wire to either end. To splice an end, follow these steps:

1. Cut the stock ECU wire with wire cutters.
2. Strip about an inch of insulation from the stock ECU wire.
3. Strip an inch of insulation from the end of the corresponding ECU+ wire.
4. Twist the two wires together and solder the connection. The two wires will be an inverted 'V' shape.
5. Bend the ECU+ wire backwards and wrap the exposed connection with electrical tape.

5.2.3 Wiring Tips

The stock wiring harness generally only exposes a few inches of wire from the stock ECU. Try to make your connections about half-way between the stock ECU connector and where it disappears under the dash.

- Make sure your hands are clean before wrapping the connection – even the slightest bit of oil will foul the adhesive on the electrical tape.
- Check and double-check that you're connecting the right wires before soldering. It's much easier to re-do a connection before soldering than after.
- A good online resource, including photos, for how to make these connections is at http://www.mmxpress.com/technical/connections.htm

5.3 Locating and Removing the Stock ECU

The procedure for removing the stock ECU depends on the type of car you're installing the ECU+ in.

For the Talon, Eclipse and Laser vehicles (all years), the stock ECU is mounted vertically and is located behind the radio in the center console. Remove both the driver and passenger kick panels, and you'll find the ECU bolted in with three bolts. Remove the bolts and the stock ECU will slide out. On the 1990-1994 cars, the stock ECU slides out on the passenger side. On the 1995-1999 cars, it slides out on the driver's side.

For the Galant VR-4, the stock ECU is bolted underneath the dash on the far right side. Remove the two bolts holding it in place, and it'll drop out.

For the Evolution VIII and IX, the stock ECU is behind the glove box. Remove the glove box (there's a tab on the left side that will free the glove box and allow it to drop out) and unbolt the two bolts holding the stock ECU in place. The stock ECU will drop straight down into the passenger side floor.

5.4 Making the Connections

With the stock ECU removed from your vehicle, it's time to do the hookup to the ECU+ wiring harnesses. All of the cars that the ECU+ supports have similar wiring, but the wire locations on the stock ECU are different.

5.4.1 1990-1994 Galant VR-4, Laser Talon and Eclipse

On the 1990-1994 Talon, Eclipse, Laser and Galant VR-4, the stock ECU contains three connectors, numbered as shown in Illustration 1. This view shows the stock ECU as you'll view it, facing the connectors in the end of the metal enclosure.

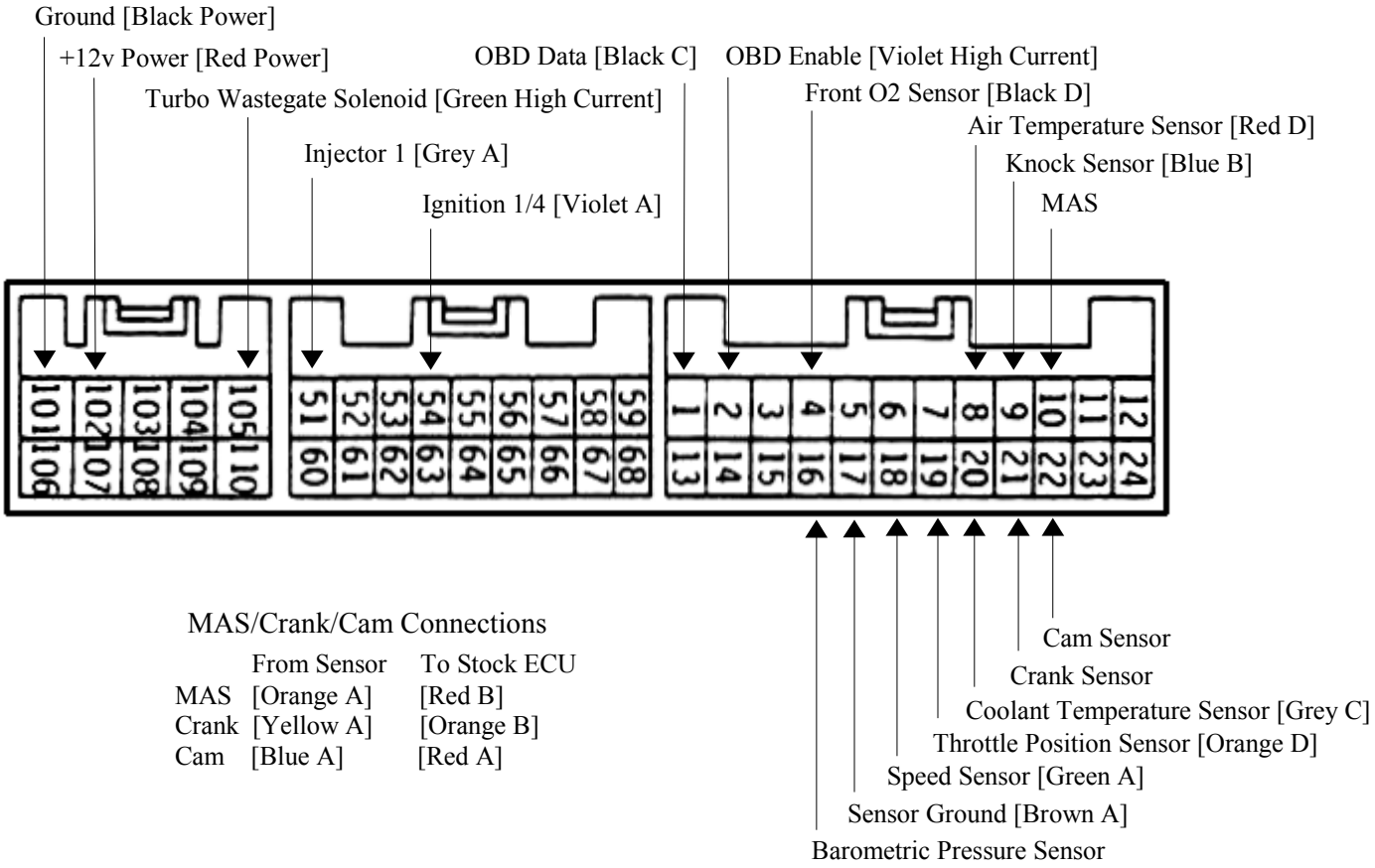


Illustration 1 - The first generation stock ECU pinout

To hook up the ECU+ head unit, you'll be wiring between the stock ECU and one of the two ECU+ harnesses. There are two ECU+ harnesses – one for "low current" connections (this is the main harness, containing 34 pins, terminated at the black connector on the head unit) and one for "high current" connections (terminates at the white connector on the head unit). The high current harness also includes the power connections.

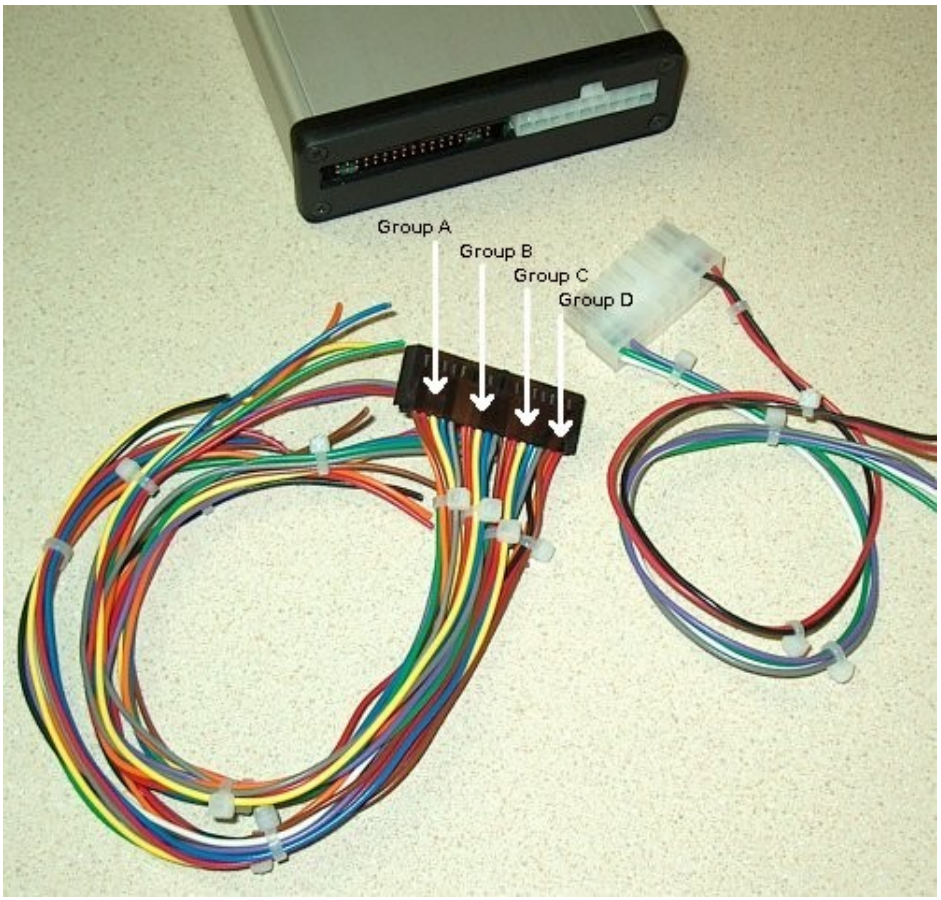


Illustration 2 - The low (left) and high (right) current harnesses.

The low current harness connector contains 34 positions arranged in groups of ten. The groups are labeled “Group A” through “Group D,” and within a wire-tied group, wires are color-coded in the following sequence: brown, black, red, orange, yellow, green, blue, violet, gray, white. In the above illustration, the Group A wires are on the left, then Group B, Group C and Group D. The leftmost brown wire in Group A is pin 1. Underneath pin 1 is the unused pin 2. The red wire beside the brown wire is pin 3, and so on. The table below lists all of the connections on the low current harness, and where they'll connect to on the stock ECU. Note that some positions on this connector are not used (like pin 2), and there is no wire in that position.

ECU+ Pin #	ECU+ Wire Color	Wiring Group	ECU+ Function	Stock ECU Pin #
1	Brown	A	Sensor Ground	17
2	(Black)	A	*Not Used*	
3	Red	A	Cam Sensor Out (to stock ECU)	22
4	Orange	A	MAS In (from sensor)	Was 10
5	Yellow	A	Crank Sensor In (from sensor)	Was 21
6	Green	A	Speed Sensor In	18
7	Blue	A	Cam Sensor In (from sensor)	Was 22
8	Violet	A	Ignition 1/4 In	54
9	Grey	A	Injector 1 In	51
10	White	A	Clutch Switch In	
11	Brown	B	External Switch In	
12	(Black)	B	*Not Used*	
13	Red	B	MAS Out (to stock ECU)	10
14	Orange	B	Crank Sensor Out (to stock ECU)	21
15	Yellow	B	Analog Output 0	
16	Green	B	Analog Output 1	
17	Blue	B	Knock Sensor In	9
18	Violet	B	Analog Output 2	
19	Grey	B	Analog Output 3	
20	(White)	B	*Not Used*	
21	Brown	C	Rear O2 Sensor In	N/A
22	Black	C	OBD Data In	1
23	Red	C	Analog Input 1/EGT Adapter In	
24	Orange	C	*Reserved*	
25	Yellow	C	External MAP Sensor In	
26	Green	C	Simulated Rear O2 Sensor Out	N/A
27	Blue	C	Analog Input 2/Wideband O2 Sensor Kit In	
28	Violet	C	Analog Input 0	
29	Grey	C	Coolant Temperature Sensor In	20
30	White	C	GM Air Temperature Sensor 0 In	
31	Brown	D	GM Air Temperature Sensor 1 In	
32	Black	D	Front O2 Sensor In	4
33	Red	D	Air Temperature Sensor In	8

ECU+ Pin #	ECU+ Wire Color	Wiring Group	ECU+ Function	Stock ECU Pin #
34	Orange	D	Throttle Position Sensor In	19

Table 1- The low current harness connections

With the exception of the MAS, crank and cam sensor connections, all of the above connections are tap connections – you "tap" the signal and feed it to the ECU+ input. For the MAS, crank and cam sensor connections, you'll split the stock ECU's wire and connect the vehicle sensor's output to the ECU+, and the corresponding ECU+ output to the stock ECU.

The high current harness connector contains 18 positions. Of this, only 6 positions are populated. The table below lists the connections on the high current harness, and where they'll connect to on the stock ECU. Note that the connections on the high current harness are numbered from right to left rather than from left to right, as on the low current harness.

ECU+ Pin #	ECU+ Wire Color	ECU+ Function	Stock ECU Pin #
10	Red	+12v Power	102
11	Black	Ground	101
8	Grey	*Reserved*	
9	White	*Reserved*	
17	Violet	OBD Enable Out	2
18	Green	Boost Control Solenoid Out	

Table 2- The high current harness connections

High current pins 10, 11 and17 are tap connections. Pin 18 is an output that doesn't connect to the stock ECU harness.

2 other connections are reserved for future use, and will be used in a future software release. The rest of the connections are not used.

Some notes:

- The 1990-1994 cars don't include a rear O2 sensor, so these connections are marked N/A in the low current table. The input marked Rear O2 Sensor In can be used as an additional 0 to 5 volt analog input, if desired.
- Some connections on both harnesses are marked *reserved*. These will be used in a future ECU+ software release.
- Any ECU+ harness connections that aren't wired up to the stock ECU or another device should be securely bundled together with electrical tape on the end to ensure that these don't

accidentally short-circuit to a in-car ground.

With the basic ECU+ installation completed, you may want to consider hooking up some of the ECU+'s other optional inputs and outputs. See the section Using the Auxiliary ECU+ Inputs and Outputs on page 18 for more information. Then skip to the next section, Finishing Up the Hardware Installation, on page 28.

5.4.2 1995 and Up Talon, Eclipse and Evolution VIII, PnP ECU+

On the 1995-1999 Talon and Eclipses, and the Evolution VIII, the stock ECU contains four connectors, numbered as shown in Illustration 3. This view shows the stock ECU as you'll view it, facing the connectors, facing the connectors in the end of the metal enclosure.

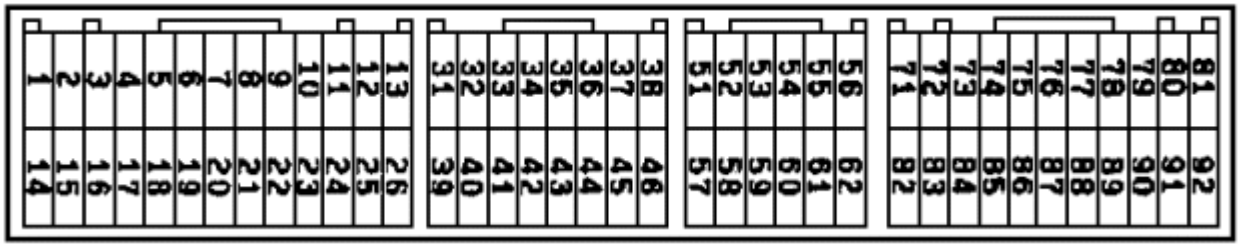


Illustration 3 - The second generation stock ECU pinout

To install the plug-n-play (PnP) version of the ECU+, simply unplug the stock connectors from the stock ECU, and plug them into the mating connector on the ECU+ wiring harness. The ECU+ wiring harness contains a similar set of connectors – plug these into the stock ECU. Each of the connectors are different sizes and keyed for orientation – it's not possible to mix them up.

Note: any ECU+ harness connections that aren't wired up to the stock ECU or another device should be securely bundled together with electrical tape on the end to ensure that these don't accidentally short-circuit to a in-car ground.

With the basic ECU+ installation completed, you may want to consider hooking up some of the ECU+'s other optional inputs and outputs. See the section Using the Auxiliary ECU+ Inputs and Outputs on page 18 for more information. Then skip to the next section, Finishing Up the Hardware Installation, on page 28.

5.4.3 1995 and Up Talon, Eclipse and Evolution VIII, non-PnP ECU+

On the 1995-1999 Talon and Eclipses, and the Evolution VIII, the stock ECU contains four connectors, numbered as shown in Illustration 4. This shows the stock ECU as you'll view it, facing the connectors in the end of the metal enclosure.

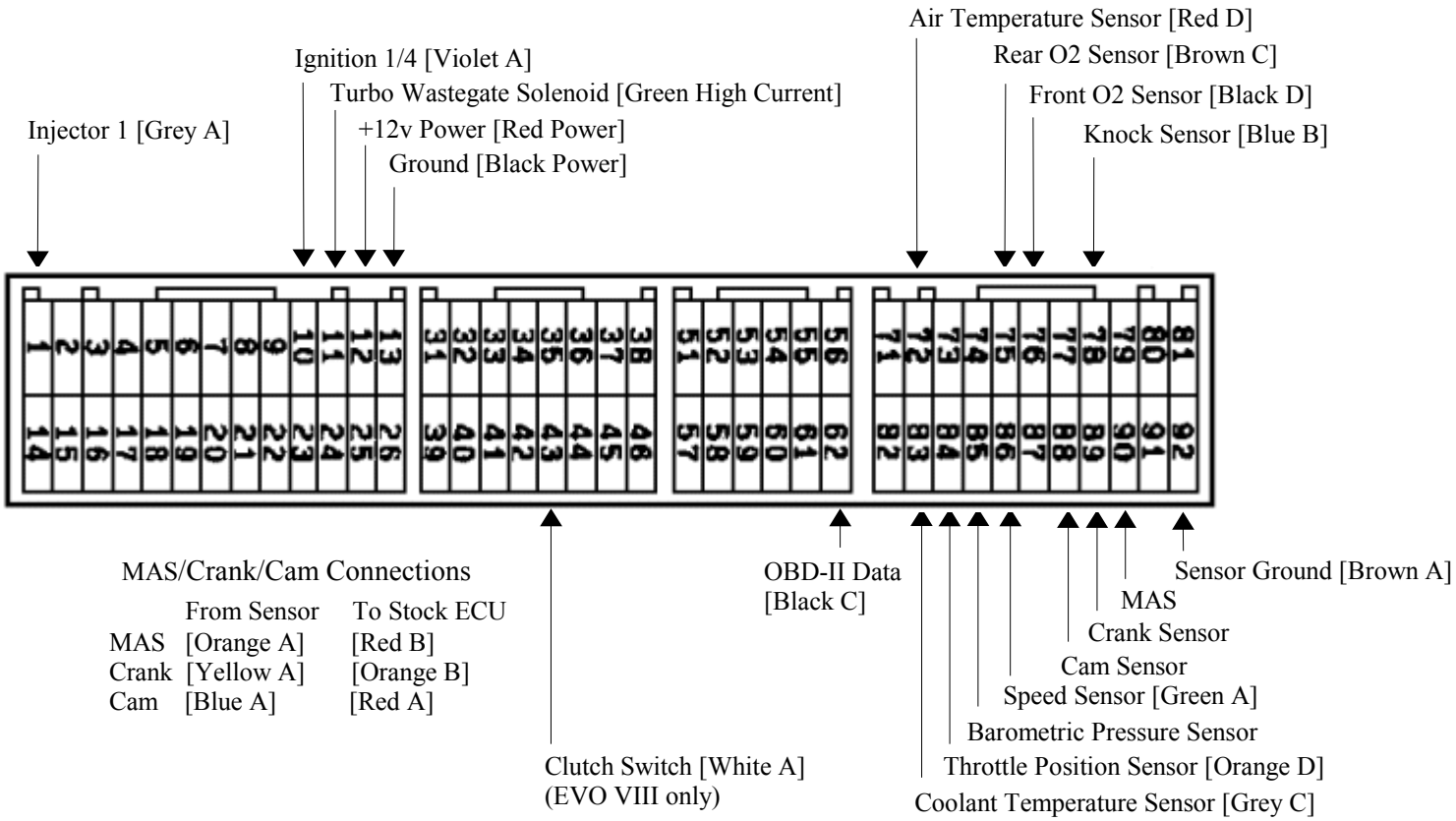


Illustration 4 - The second generation DSM and Evolution VIII ECU pinout

To hook up the ECU+ head unit, you'll be wiring between the stock ECU and one of the two ECU+ harnesses. There are two ECU+ harnesses – one for "low current" connections (this is the main harness, containing 34 pins, terminated at the black connector on the head unit) and one for "high current" connections (terminates at the white connector on the head unit). The high current harness also includes the power connections.

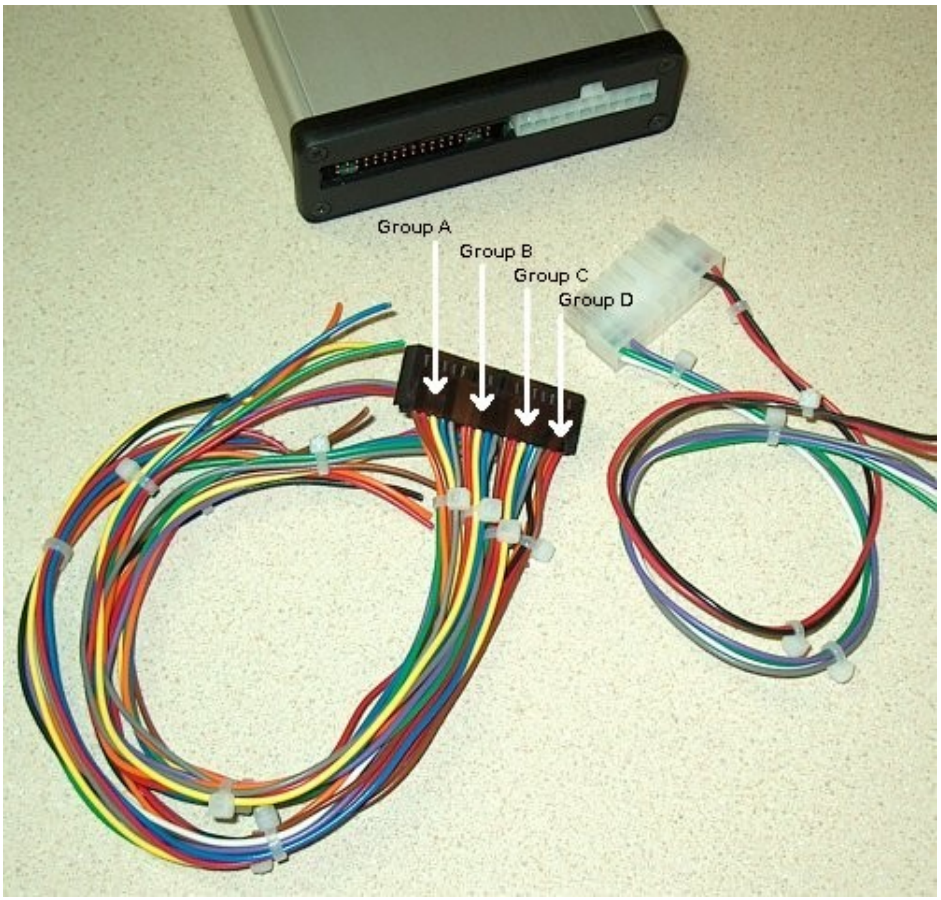


Illustration 5 - The low (left) and high (right) current harnesses.

The low current harness connector contains 34 positions arranged in groups of ten. The groups are labeled “Group A” through “Group D,” and within a wire-tied group, wires are color-coded in the following sequence: brown, black, red, orange, yellow, green, blue, violet, gray, white. In the above illustration, the Group A wires are on the left, then Group B, Group C and Group D. The leftmost brown wire in Group A is pin 1. Underneath pin 1 is the unused pin 2. The red wire beside the brown wire is pin 3, and so on. The table below lists all of the connections on the low current harness, and where they'll connect to on the stock ECU. Note that some positions on this connector are unused (like pin 2), and there is no wire in that position.

ECU+ Pin #	ECU+ Wire Color	Wiring Group	ECU+ Function	Stock ECU Pin #
1	Brown	A	Sensor Ground	92
2	(Black)	A	*Not Used*	
3	Red	A	Cam Sensor Out (to stock ECU)	88
4	Orange	A	MAS In (from sensor)	Was 90
5	Yellow	A	Crank Sensor In (from sensor)	Was 89
6	Green	A	Speed Sensor In	86
7	Blue	A	Cam Sensor In (from sensor)	Was 88
8	Violet	A	Ignition 1/4 In	10
9	Grey	A	Injector 1 In	1
10	White	A	Clutch Switch In	43 on Evolution VIII
11	Brown	B	External Switch In	
12	(Black)	B	*Not Used*	
13	Red	B	MAS Out (to stock ECU)	90
14	Orange	B	Crank Sensor Out (to stock ECU)	89
15	Yellow	B	Analog Output 0	
16	Green	B	Analog Output 1	
17	Blue	B	Knock Sensor In	78
18	Violet	B	Analog Output 2	
19	Grey	B	Analog Output 3	
20	(White)	B	*Not Used*	
21	Brown	C	Rear O2 Sensor In	75
22	Black	C	OBD-II Data In	62
23	Red	C	Analog Input 1/EGT Adapter In	
24	Orange	C	*Reserved*	
25	Yellow	C	External MAP Sensor In	
26	Green	C	Simulated Rear O2 Sensor Out	
27	Blue	C	Analog Input 2/Wideband O2 Sensor Kit In	
28	Violet	C	Analog Input 0	
29	Grey	C	Coolant Temperature Sensor In	83
30	White	C	GM Air Temperature Sensor 0 In	
31	Brown	D	GM Air Temperature Sensor 1 In	
32	Black	D	Front O2 Sensor In	76

ECU+ Pin #	ECU+ Wire Color	Wiring Group	ECU+ Function	Stock ECU Pin #
33	Red	D	Air Temperature Sensor In	72
34	Orange	D	Throttle Position Sensor In	84

Table 3- The low current harness connections

With the exception of the MAS, crank and cam sensor connections, all of the above connections are tap connections – you "tap" the signal and feed it to the ECU+ input. For the MAS, crank and cam sensor connections, you'll split the stock ECU's wire and connect the vehicle sensor's output to the ECU+, and the corresponding ECU+ output to the stock ECU.

The high current harness connector contains 18 positions. Of this, only 6 positions are used. The table below lists the connections on the high current harness, and where they'll connect to on the stock ECU. Note that the connections on the high current harness are numbered from right to left rather than from left to right, as on the low current harness.

ECU+ Pin #	ECU+ Wire Color	ECU+ Function	Stock ECU Pin #
10	Red	+12v Power	12
11	Black	Ground	13
8	Grey	*Reserved*	
9	White	*Reserved*	
17	Violet	*Reserved*	
18	Green	Boost Control Solenoid Out	

Table 4- The high current harness connections

High current pins 10 and 11 are tap connections. Pin 18 is an output that doesn't connect to the stock ECU harness.

3 other connections are reserved for future use, and will be used in a future software release. The rest of the connections are not used.

Some notes:

- Some connections on both harnesses are marked *reserved*. These will be used in a future ECU+ software release.
- Any ECU+ harness connections that aren't wired up to the stock ECU or another device should be securely bundled together with electrical tape on the end to ensure that these don't accidentally short-circuit to a in-car ground.

With the basic ECU+ installation completed, you may want to consider hooking up some of the ECU+'s

other optional inputs and outputs. See the section Using the Auxiliary ECU+ Inputs and Outputs on page 18 for more information. Then skip to the next section, Finishing Up the Hardware Installation, on page 28.

5.4.4 2006 and Up Evolution IX, PnP ECU+

On the 2006 and up Evolution IX, the stock ECU contains three connectors, numbered as shown in Illustration 6. This view shows the stock ECU as you'll view it, facing the connectors, facing the connectors in the end of the metal enclosure.

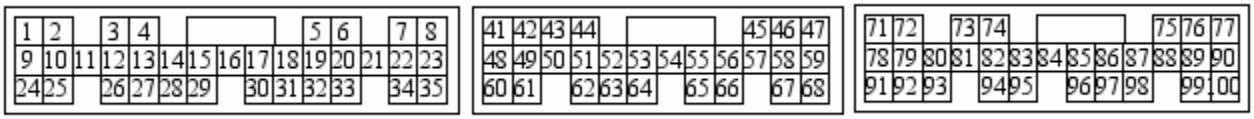


Illustration 6 - The Evolution IX stock ECU pinout

To install the plug-n-play (PnP) version of the ECU+, simply unplug the stock connectors from the stock ECU, and plug them into the mating connector on the ECU+ wiring harness. The ECU+ wiring harness contains a similar set of connectors – plug these into the stock ECU. Each of the connectors are different sizes and keyed for orientation – it's not possible to mix them up.

Note: any ECU+ harness connections that aren't wired up to the stock ECU or another device should be securely bundled together with electrical tape on the end to ensure that these don't accidentally short-circuit to a in-car ground.

With the basic ECU+ installation completed, you may want to consider hooking up some of the ECU+'s other optional inputs and outputs. See the section Using the Auxiliary ECU+ Inputs and Outputs on page 18 for more information. Then skip to the next section, Finishing Up the Hardware Installation, on page 28.

5.5 Using the Auxiliary ECU+ Inputs and Outputs

The ECU+ contains a number of additional inputs and outputs that add value to a standard installation. Included are a simulated rear O2 sensor output and several analog inputs and outputs. These are covered below.

5.5.1 The Simulated Rear O2 Sensor Output

Note: This section is not applicable to 1990-1994 DSMs, as those vehicles only include a front O2 sensor.

The 1995 and up Talon, Eclipse and Evolution VIII/IX vehicles include a rear O2 sensor on the far side of the catalytic converter. The stock ECU checks this signal to verify that the vehicle has a functional catalytic converter. If the cat is removed for race purposes, the rear O2 sensor signal won't respond as the stock ECU expects, and will often trigger a “check engine” light to flag the problem. The ECU+ head unit includes a simulated rear O2 output (for off-road use only) that can eliminate this problem. The signal simulates the signal that the stock ECU will see from a functional rear O2 sensor, and thus avoids the check engine light.

To connect the simulated rear O2 signal, locate pin 26 (green, Group C – see Table 3) on the ECU+'s low current harness. Then locate pin 75 on the stock Talon, Eclipse or Evolution VIII ECU (see Illustration 4), or pin 73 on the Evolution IX ECU (see Illustration 6). It should be tapped with a brown

wire to the ECU+ head unit's rear O2 sensor input. Cut the stock ECU wire between the stock ECU's male connector and where it splices into the brown wire, and solder in the simulated rear O2 signal to the wire containing the splice. This feeds the ECU+'s simulated rear O2 sensor output to the stock ECU, and eliminates the check engine light.

Note: in the previous paragraph, if you've purchased a PnP ECU+, the connection should be made in the ECU+'s PnP harness, not in the stock ECU wiring.

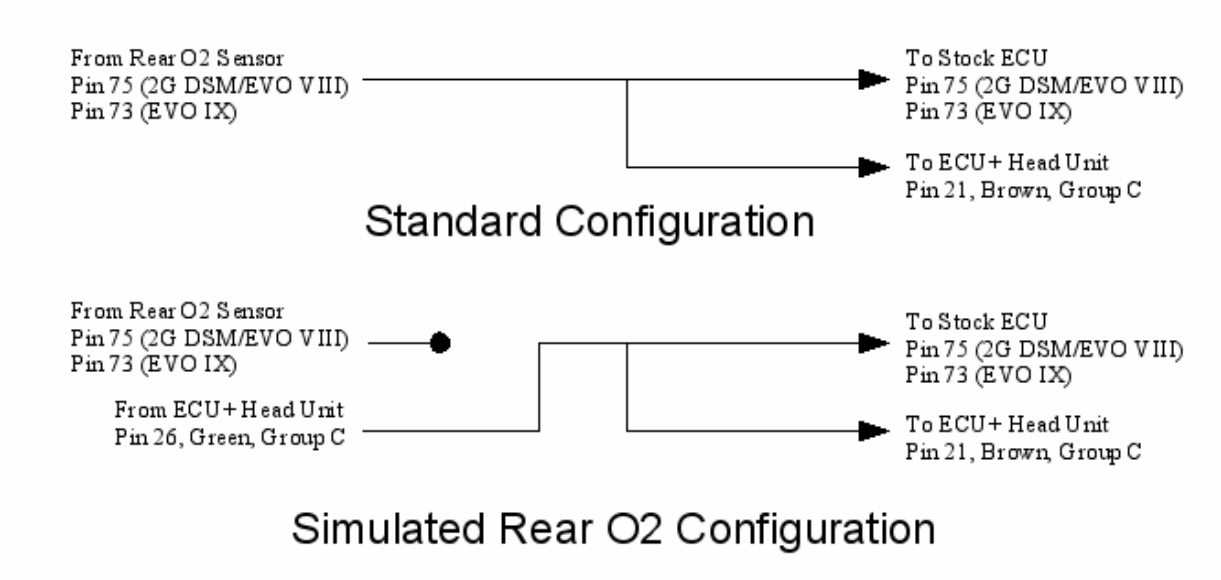


Illustration 7 - The analog wiring changes for the simulated rear O2 sensor signal

5.5.2 Analog Inputs

The ECU+ includes three uncommitted analog inputs whose voltage is logged along with all of the other ECU+ inputs. These are high impedance (2M ohm resistance) inputs that can accept any signal between 0 and 5 volts. Many aftermarket vehicle sensors can be configured to produce a “5 volt” output, and you can datalog these values with the ECU+.

Warning: before connecting a sensor input to an ECU+ analog input, verify (using a digital multimeter) that the sensor doesn't generate less than zero or greater than 5 volts. Connecting an out-of-range sensor can damage or destroy your ECU+ head unit.

To connect up a 5 volt input, simply wire it directly to the appropriate ECU+ input. Analog input 0 is pin 28 (violet wire, Group C) on the ECU+, analog input 1 is pin 23 (red wire, Group C) and analog input 2 is pin 27 (blue wire, Group C).

Analog inputs 1 and 2 also support special modes of operation. Analog input 1 can be used as the input from a 5 volt EGT adapter, while analog input 2 can be used as the input from a wideband O2 sensor kit. See Using a EGT Sensor and Adapter on page 23 and Using a Wideband O2 Sensor Kit on page 21, respectively, for more information.

5.5.3 GM Air Intake Temperature Sensor Inputs

Pins 30 (white wire, group C) and 31 (brown wire, group D) of the ECU+'s low current connector are reserved for connecting up to two GM AIT sensors. Values read from these sensors can be datalogged alongside the other ECU+ inputs. Additionally, the first sensor can be used to simulate the function of

the stock air temperature sensor when the ECU+ is being used with a GM MAF. See the section Analog Outputs on page 97 for more information about configuring the software to support these inputs, and the section Custom Logging Configuration on page 44 to configure the software to datalog these values.

To connect up a GM AIT sensor, first purchase the sensor from a GM dealer or from a variety of online vendors such as GM Parts Direct. The AIT part number is 25036751, and you'll also need the pigtail harness (part number pig tail 12102620). The pigtail has two (interchangeable) wires; connect one to ground and the other to the ECU+'s pin 30 or 31.

5.5.4 Analog Outputs

The ECU+ includes four uncommitted analog outputs that can generate any fixed voltage between 0 and 5 volts under software control.

To connect up the ECU+'s analog outputs, simply wire it to another device (or an ECU+ analog input) as desired. Analog output 0 is pin 15 (yellow wire, group B), analog output 1 is pin 16 (green wire, group B), analog output 2 is pin 18 (violet wire, group B) and analog output 3 is pin 19 (grey wire, group B). Note that these analog outputs aren't fully buffered on the ECU+, and connecting them directly to ground or the output of another device may damage your ECU+ head unit. It's recommended that any device that connects to these outputs have an input impedance of 10k ohms or greater.

Analog outputs 0 and 1 are typically used when the ECU+ is used with a GM MAF meter. See the section Using the ECU+ with a GM MAF Meter on page 132 for more information.

Analog output 2 can be used to generate a simulated narrowband O2 output that's derived from the signal generated by a wideband O2 sensor kit. See the section Using the Simulated Front O2 Sensor Output on page 23 for more information.

5.5.5 Using a MAP Sensor

The ECU+ directly supports several aftermarket MAP sensors using a dedicated input. A MAP (Manifold Absolute Pressure) sensor allows the ECU+ to directly datalog and read out your car's boost and vacuum level.

- The ECU+ currently includes calibration tables for five specific MAP sensors:
- The “GM 3-Bar” map sensor (part number 12223861), available from any GM dealer or GM Parts Direct (http://www.gmpartsdirect.com).
- The SenSym ASCX30 two-bar map sensor, available from DigiKey (http://www.digikey.com).
- The AEM 3.5 Bar map sensor, available from many AEM dealers.
- The AEM 5 bar map sensor.
- The Freescale MPX4250AP map sensor, also available from Digikey.

In addition to the above sensors, the ECU+ can support *any* linear 5 volt MAP sensor by using the “custom MAP sensor” support built into the ECU+ Win software.

In general, a MAP sensor has a port for pressure, and at least three electrical connections, with one terminal for a +5 volt power source, one for ground, and one for the output voltage. Install the sensor's pressure input just like you would a boost gauge, and then connect a long wire from the sensor output in the engine bay, through the firewall to your ECU+'s MAP sensor input pin (pin 25, yellow wire, Group C). You'll need to locate a +5 volt source under-hood for the sensor, which is available by tapping the fixed 5 volt side of any of several of the stock engine sensors (most commonly, the throttle position

sensor).

For the GM sensor, make the electrical connections as follows:

- 'A' terminal to ground
- 'B' terminal is the sensor output
- 'C' terminal to a +5v voltage source

The SenSym sensor has six electrical pins. With the sensor laying flat on a table (pressure port facing up, electrical pins at the bottom), the pins are numbered 1-6 from left to right. Make the electrical connections as follows:

- Pin 2 to a +5v voltage source
- Pin 3 is the sensor output
- Pin 4 to ground

The AEM sensors have three electrical connections. Looking into the end of the sensor, with the tab at the top, the connections are as follows:

- Upper right pin to a +5v voltage source (red wire)
- Upper left pin to ground (black wire)
- Bottom pin is the sensor output (green wire)

The Freescale sensor has six electrical pins. Pin 1 is notched, and pins are numbered 1-6 from left to right. Make the electrical connections as follows:

- Pin 1 is the sensor output
- Pin 2 to ground
- Pin 3 to a +5v voltage source

Be very careful when installing a MAP sensor. Use a digital multimeter to verify your +5 volt source, and under no circumstances connect the MAP sensor to +12 volts. Doing so may damage the sensor and/or your ECU+ head unit.

Once the MAP sensor is installed and connected to the ECU+, configure the ECU+ software so that it knows what MAP sensor is installed. See section MAP Sensor on page 94 for more information.

5.5.6 Using a Wideband O2 Sensor Kit

The ECU+ supports a number of wideband O2 sensor kits. A wideband O2 sensor "kit" uses an off-the-shelf wideband oxygen sensor along with some electronics to control the sensor heater and to condition and linearize the output of the sensor. When a supported kit is connected to the ECU+, the ECU+ can capture and display a real-time graph of your engine's air/fuel ratio. This is *very* useful when adjusting fuel with the ECU+.

The ECU+ Win software supports the following wideband kits:

- PLX Devices M-Series (M-200, M-250, M-300, M-400 and M-500) Wideband Oxygen Sensor Controllers, with or without the M-Series gauges. See http://www.plxdevices.com for more information.
- FJO Racing Wide Band Oxygen Sensor kits using controller CWC0002 or CWC0001. See http://www.fjoracing.com.
- Innovate Motorsports LM-1 or LC-1 Air/Fuel Gauge. See http://www.innovativemotorsports.com.

- Tech Edge WB02 Lambda Meters (WBo2 version 2A0 and TE-WB Wideband Unit (v1.5)). See http://www.techedge.com.au.
- Zeitronix Zt-2 Wideband Air/Fuel Ratio Meter. See http://www.zeitronix.com.
- AEM Wideband UEGO Controller and Gauge-Type Wideband UEGO Controller. See http://www.aempower.com.
- Auto Meter Wideband Air and Fuel Ratio Meter. See http://autometer.com
- The NGK Powerdex AFX Air Fuel Monitor. See http://ngksparkplugs.com

To use a wideband kit with the ECU+, first install the device as described by the manufacturer, to include any calibration sequences. Next, connect the output of the wideband controller to the analog input 2 pin of the ECU+. This is pin 27 (blue wire, Group C) on the ECU+ head unit. Then configure the software so that the ECU+ knows which wideband kit you're using (see the section Miscellaneous Configuration on page 53).

Connecting up a wideband is usually pretty straightforward. The main difference between the the supported kits lie in which wideband output you should connect to the ECU+ head unit:

- For the PLX M-Series, use the WB output (white wire). This output is 0 volts for 10:1 AFR and 5 volts for 20:1 AFR.
- For the FJO kits, use the red analog output wire on the black 4-pin square connector. The FJO's use a non-linear voltage to AFR mapping.
- For the Innovate LM-1, a 3.5mm stereo connector provides the output that the ECU+ needs. Pin 2, the connector ring, is analog output 2. Connect this to the ECU+ head unit. This output is 1 volt for 10:1 AFR and 2 volts for 20:1 AFR.
- For the Innovate LC-1, use the brown analog out 2 wire. This output is 0 volts for 7.35:1 AFR and 5 volts for 22.39:1 AFR.
- For the Tech Edge 2A0, use the WBlin output, which is pin 4 on the unit's RJ45 connector. Facing the RJ45 connector, the pins are numbered 1-8 from left to right. The WBlin output is 9:1 AFR at 0 volts 19:1 AFR at 5 volts.
- For the Tech Edge TE-WB 1.5, use the WBlin output, which is pin 4 on the unit's male DB-9 connector. Facing the DB-9, the top row of pins are numbered 1-5 from left to right. The WBlin output has a non-linear voltage to AFR mapping.
- For the Zt-2, use the Wideband Analog Output. This is the white wire on pin 2. This output uses a non-linear voltage to AFR mapping.
- For the AEM Wideband UEGO Controller, use the orange wire, pin 1 (Sensor #1, 0-5v). This output uses a non-linear voltage to AFR mapping.
- For the AEM Gauge-Type Wideband UEGO Controller, use the white wire, 0-5v output. This output uses a non-linear voltage to AFR mapping.
- For the Auto Meter Gauge, use the blue wire, 0-4v output. Ensure that the output is 10:1 AFR for 0 volts, 16:1 AFR for 4 volts.
- For the NGK Powerdex AFX, use the yellow wire. This output is 9:1 AFR at 0 volts, and 16:1 AFR at 5 volts.

For any of the widebands, be sure that the ground for the controller is identical to the ground on the ECU+ head unit.

In general, you shouldn't modify any of the factory settings on it with respect to the analog output "curve," as ECU+ is pre-configured to work with the factory output voltages on all of the supported

widebands. However, for the Innovate LM-1 and LC-1 units it is possible to get more accurate AFR readings by modifying the output curve. To do so:

- Using Innovate's LM-Programmer software, modify the analog out 2's curve so that 0 volts corresponds to 10:1 AFR and 5 volts corresponds to 20:1 AFR.
- Configure the ECU+ software as if you were using the PLX M-Series devices.

This configuration gives you the higher-resolution PLX curve from the Innovate device, resulting in more accurate AFR readings.

5.5.7 Using a EGT Sensor and Adapter

The ECU+ supports datalogging of engine exhaust gas temperature (EGT) using a external EGT probe and either the "5V K-type EGT adapter" from o2simulator.com (see http://www.o2simulator.com/AEMEGTADAPTER.htm) or the PLX Exhaust Gas Temperatur Sensor Module (see http://www.plxdevices.com) . These adapters convert the signal from a K-Type thermocouple EGT probe into a 0-5v signal which the ECU+ can datalog as actual exhaust gas temperature.

To use one of these adapters, first install the adapter and EGT probe according to the manufacturers instructions. EGT probes typically are screwed into the exhaust manifold. Next, connect the analog output wire from the adapter to the analog input 1 pin of the ECU+ head unit. This is pin 23, red, group C on the ECU+ wiring harness. Finally, configure the ECU+ Win software to display actual EGT temperature from analog input 1 rather than the default voltage. The ECU+ is pre-configured to work with either of these adapters to display actual exhaust temperatures.

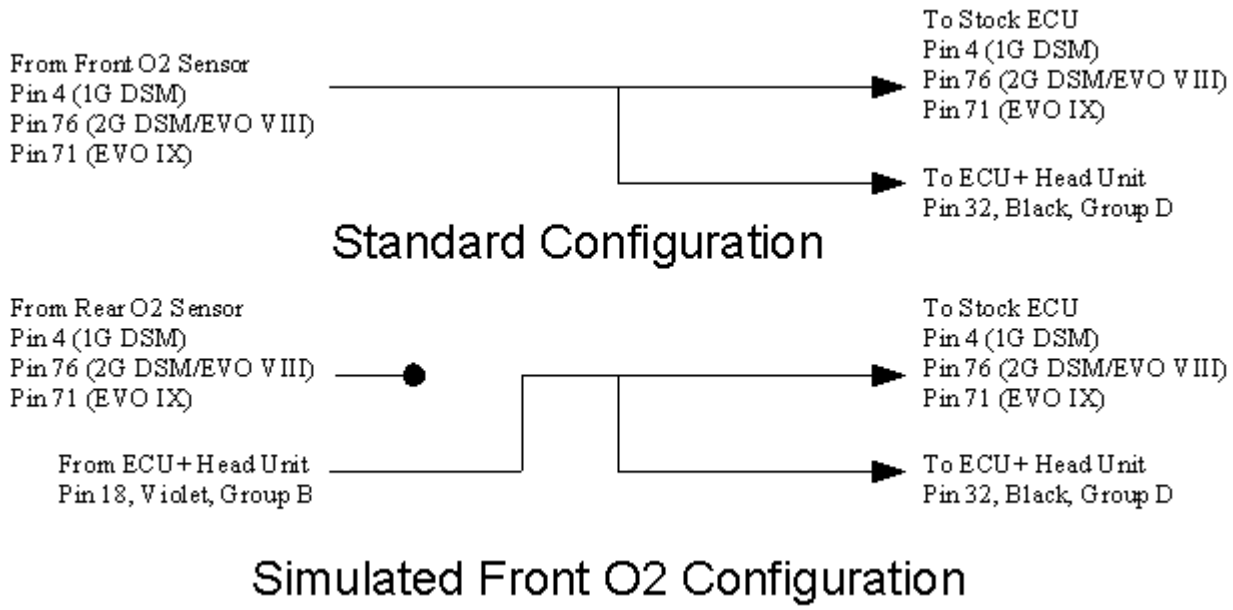
5.5.1 Using the Simulated Front O2 Sensor Output

In addition to the simulated rear O2 sensor output (described in the section The Simulated Rear O2 Sensor Output on page 18), the ECU+ can also generate a simulated *front* O2 sensor signal. The front O2 sensor on all of the supported vehicles is the stock ECU's primary feedback mechanism when running at idle and low load situations. Without a valid front O2 sensor signal, the vehicle will run poorly and generate multiple check-engine codes.

The purpose of the simulated front O2 sensor feature of the ECU+ is to allow customers with a supported wideband O2 sensor kit to install their wideband O2 sensor in the downpipe in the "bung" normally used by the stock front O2 sensor. This configuration saves money, in that no extra drilling or welding is required to install the wideband O2 sensor.

To connect the simulated front O2 signal, locate pin 18 (violet, Group B – see Table 3) on the ECU+'s low current harness. Then locate pin 4 on the stock 1990-1994 Laser, Talon or Eclipse ECU (see Illustration 1), pin 76 on the stock 1995 and up Talon, Eclipse or Evolution VIII ECU (see Illustration 4), or pin 71 on the Evolution IX ECU (see Illustration 6). It should be tapped with a black wire to the ECU+ head unit's front O2 sensor input. Cut the stock ECU wire between the stock ECU's male connector and where it splices into the black wire, and solder in the simulated front O2 signal to the wire containing the splice. This feeds the ECU+'s simulated front O2 sensor output to the stock ECU.

Note: in the previous paragraph, if you've purchased a PnP ECU+, the connection should be made in the ECU+'s PnP harness, not in the stock ECU wiring.



With the simulated front O2 sensor output connected up, you'll need to configure the ECU+'s analog output 2 to generate the simulated signal. See the section Analog Outputs on page 97 for more information.

Note that if you actually remove the stock front O2 sensor (rather than leaving it installed and tied off somewhere), the stock ECU may throw a check-engine code because it doesn't see any load on the front O2 sensor heater wire. To work around this, a 15 ohm, 30 watt resistor installed on the heater wire will fool the stock ECU into thinking that a real front O2 sensor is installed. Such a resistor is part number LT30J-15-ND, from DigiKey (http://www.digikey.com). You may need a heat sink on the resistor, as it gets quite hot.

5.5.2 The Clutch Switch Input

To implement the no-lift-to-shift (NLTS) feature, the ECU+ includes an input that allows the unit to monitor the clutch pedal position. The ECU+ uses this "clutch switch" input to determine when to activate the NLTS algorithm.

The clutch switch is a feature of all of the vehicles supported by the ECU+. The clutch switch is a plunger-type switch that's installed on the clutch pedal near where the clutch hinge is. This switch is activated by light pressure on the clutch, and is sometimes known as the "top clutch switch." (Auto-transmission cars don't have this switch, obviously.) When the clutch is released, this switch generates a +12 volt signal, when pressed, the clutch switch reads ground.

Connecting this switch to the ECU+ is a function of the vehicle type that the ECU+ is installed in. In each case, the wire at pin 10 (white wire, Group A) on the ECU+'s low current harness must be connected to the clutch switch. The connections are as follows:

- For the Evolution IX vehicles, the ECU+ connection is made at the factory.
- For the Evolution VIIIs with a plug-n-play harness, the connection is made at the factory.
- For the wire-it-yourself Evolution VIII model of the ECU+, a tap connection must be made in the stock ECU wiring at the stock ECU pin 43.
- For the 1990-1999 Talon, Eclipse, Laser and Galant VR4s, a connection must be manually made to the physical clutch switch under the dash. To do so,

- First locate the switch as shown below. Illustration 9 shows the under-dash view on a first generation DSM.

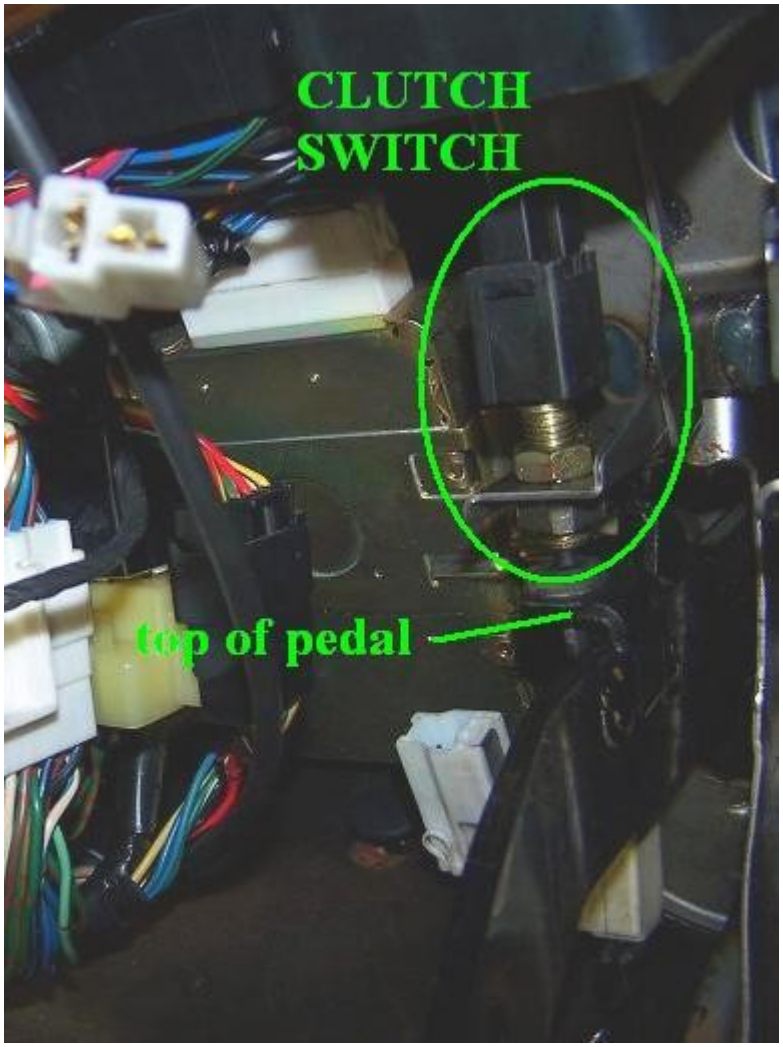


Illustration 9: The clutch switch on both generations of DSMs

- One one side of the connector is a black wire (with a white stripe). Verify, using a digital multimeter, that this wire is +12 volts when the clutch is out, or ground when the clutch is pressed.
- Next, remove the connector from the switch. See Illustration 10.

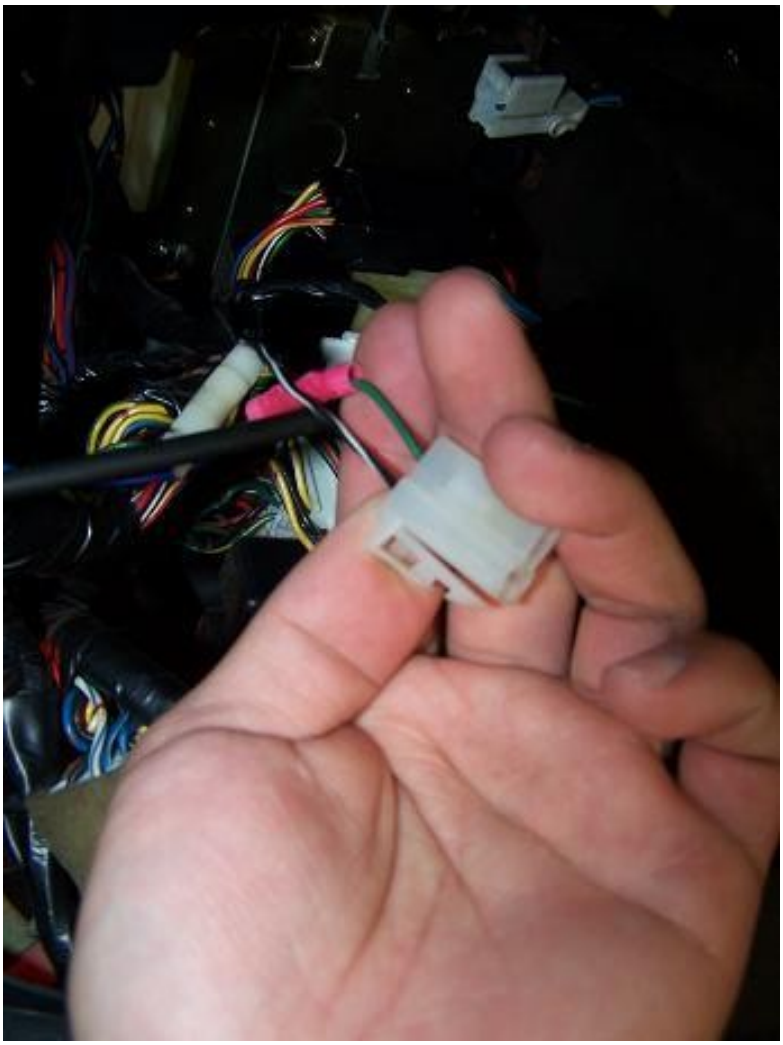


Illustration 10: The clutch switch connector.

- Tap the black wire and connect it to the white, group A wire on the ECU+ low current harness.

Note: when connecting to the clutch switch on the DSMs, always verify the voltages on the switch before connecting it to the ECU+. Send corrections to this procedure to support@ecuplus.com.

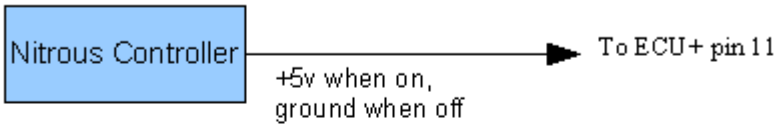
5.5.3 The External Switch Input

The ECU+ supports an "external switch" input that can be used to determine which set of fuel and timing maps the ECU+ is using at any given time. (In the future, this input may be used for other applications.) This input is sampled 50 times per second, and can instantaneously alter which set of maps is in use.

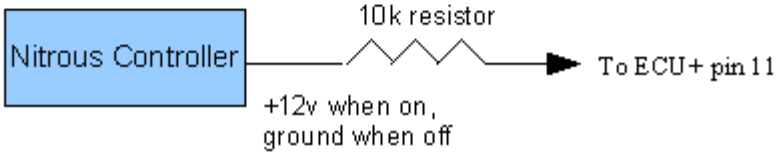
This switch input is internally "pulled up" with a resistor to 5 volts. If left disconnected, the input is at +5 volts. To use the input, connect an external device to pin 11 on the black low current harness (brown wire, Group B). Three sample configurations are shown:

- The external device (like a nitrous controller) generates a 5 volt signal or a ground signal to indicate its status. This device can be used directly.
- The external device generates a higher-than-5 volt signal or a less-than-ground signal. To use this device, connect its output through a 10k resistor to the ECU+ input. If the series resistor is

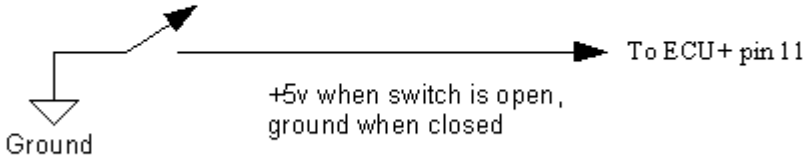
not used, the high voltage may damage the ECU+.



Sample Configuration for a +5v Device



Sample Configuration for a +12v Device



Sample Configuration for a Toggle Switch

- The external device is a simple SPST (single pole, single throw) or SPDT (single pole double throw) toggle switch, with the common (center pin) grounded. This can be used directly, as the ECU+ includes an internal pull-up resistor.

5.5.4 Connecting a Boost Control Solenoid

The ECU+ can control boost using a GM boost control solenoid. This device is installed as follows:

- Purchase the appropriate solenoid. This is GM part number 1997152. Also purchase the associated pigtail harness, part number 12102747.
- The solenoid has three connections:

  Port 1: Silver plug, by itself on one end.
  Port 2: Opposite port, next to port 3.
  Port 3: Comes with a small filter on it.
- Electrically, both wires are equivalent. Connect one to the ECU+, pin 18, green wire, on the high-current connector. Connect the other wire to a switched +12v connection. Use a voltmeter to verify that the +12v connection is switched - +12 volts when the vehicle's ignition is on, 0 volts when it's off.
- Next, use vacuum hoses to connect the solenoid to your turbocharger, as follows:
  - Solenoid port 1 (the silver one) to a source of boost pressure. This is usually connected to the nipple on the turbo's compressor cover or to a tap on the upper intercooler pipe.
  - Solenoid port 2 to the turbo's wastegate actuator port.

  Zip-tie all vacuum hoses very tightly.

5.6 Finishing Up the Hardware Installation

When you're all done making the ECU+ connections, you'll have some unused wires in both the low current and high current connectors. Tie these wires together with wire ties and wrap them in electrical tape. Next, secure all of the ECU+ harness wire bundles by applying wire ties to each bundle. In particular, be sure to use a wire tie very near each of the connectors at the head unit – this will ensure that there's no stress on the pins in the cables.

For the non-PnP ECU+, you'll want to verify that you've wired the ECU+ power correctly. Using a multimeter, hook up to the ECU+ two-pin power connection and turn your car's ignition to the on position. Your multimeter should read between **positive** 12 and 15 volts. If not, double-check your connections. With the wiring completed, you should plug the ECU+ head unit into the newly-wired harnesses.

Next, connect the included 9-pin serial cable to the appropriate connector on the ECU+ head unit.

Before starting your car for the first time, install the ECU+ software and follow the instructions in the next few sections to use the Windows software to setup the ECU+ head unit. In particular, be sure that the cam and crank angle sensor types are set appropriately for your car. Once everything is configured correctly, turn off your ignition and leave it off for a few seconds, then start your vehicle. The vehicle should run normally, just like it did before you installed the ECU+. If not, see the next section for troubleshooting information.

5.7 Troubleshooting the Hardware Installation

In the previous section, you installed your ECU+ head unit. Here are some things to check if your car doesn't run normally after the install.

1. For all problems, double- and triple-check the wiring. Un-tape the connections and look at them. Is a good solder joint being made? Are the wires connected according to the tables?
2. If your car won't start or idles very poorly and refuses to rev, chances are that either the cam or crank angle sensor signals aren't working. Verify that the head unit is using the proper signal type. As a last resort, remove the ECU+ head unit from its connectors and manually jumper together the cam and crank sensor signals.
3. If the car idles very fast and seems to be running rich, the MAS signal is probably not being regenerated. Again, check the wiring. Does the software indicate that the input MAS signal is 0 Hz? If so, the MAS input may be disconnected. If not, the MAS output may be disconnected.

As a last resort, contact your ECU+ dealer and see what suggestions they might have. The ECU+ is tested at the factory, and should work for your car.

6 Software Installation

The ECU+ system contains several pieces of software which need to be installed:

- The ECU+ Win software, which is a graphical Microsoft Windows-based display, tuning and analysis tool.
- An alternate version of ECU+ Win which provides stock ECU ROM editing and map tracing for use with the EcuFlash application. This is optional.

To install the software, insert the CD-ROM supplied with the system into your CD-ROM drive. Then use the Windows Explorer tool to navigate to the CD-ROM, and double-click the Setup.exe program.

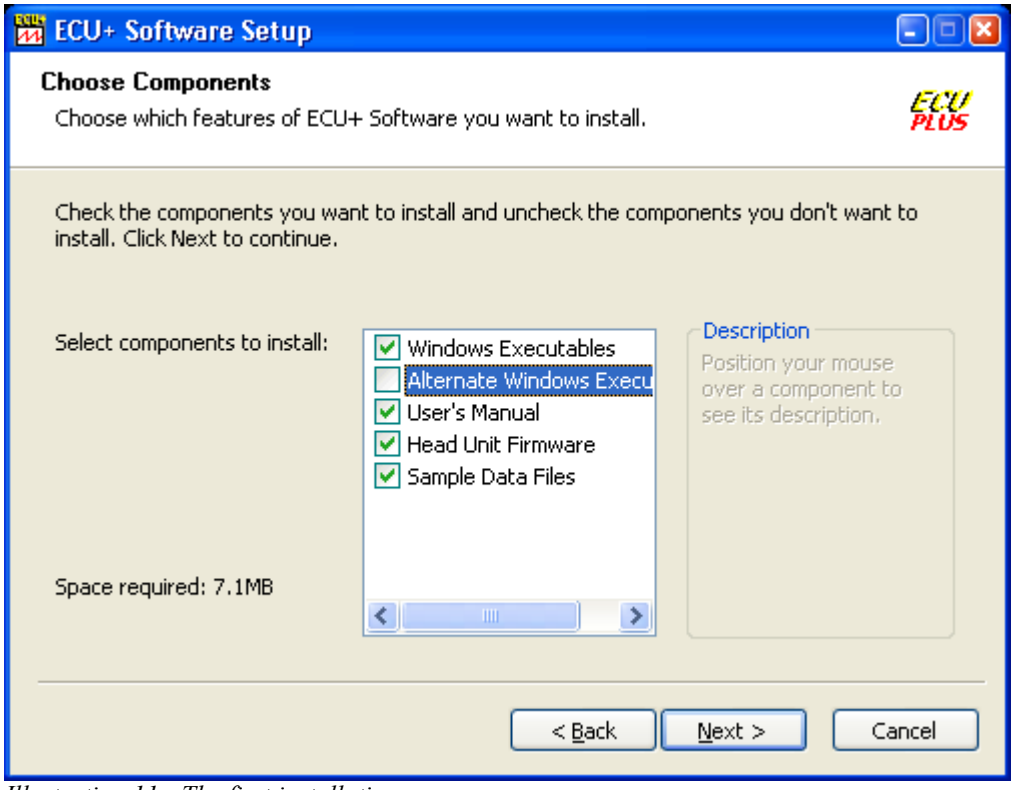
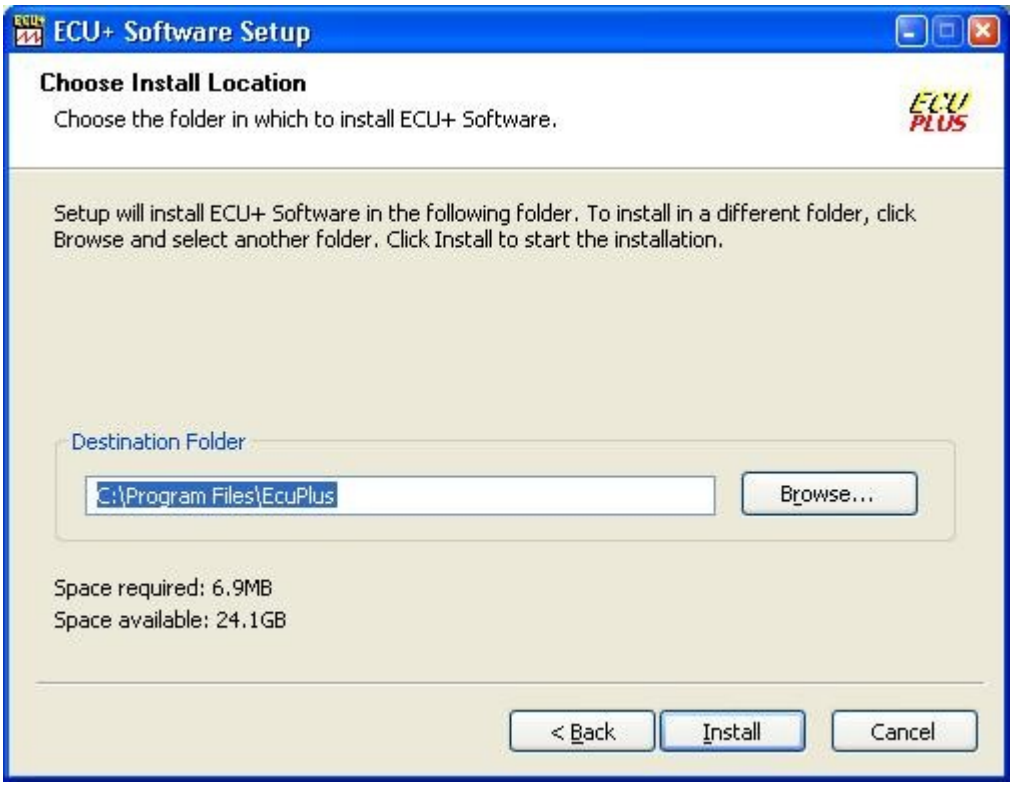


Illustration 11 - The first installation screen

The ECU+'s graphical install will begin, and, after accepting the license agreement, you'll be prompted for which pieces of the software suite to install.

In general, you'll want to install the Windows Executables, User's Manual and Sample Data Files. The Alternate Windows Executables option installs the version of the ECU+ Windows software that also includes stock ECU ROM editing and map tracing. Install this if you'd like the ECU+ to work alongside the EcuFlash program.

Next, the installer will prompt you for a directory to install to.



For compatibility with other Windows installations, most people just select the default.

Finally, the installer will copy the ECU+ files to the appropriate location, update the Windows registry, and install Start menu and desktop shortcuts. A successful screen looks like this:

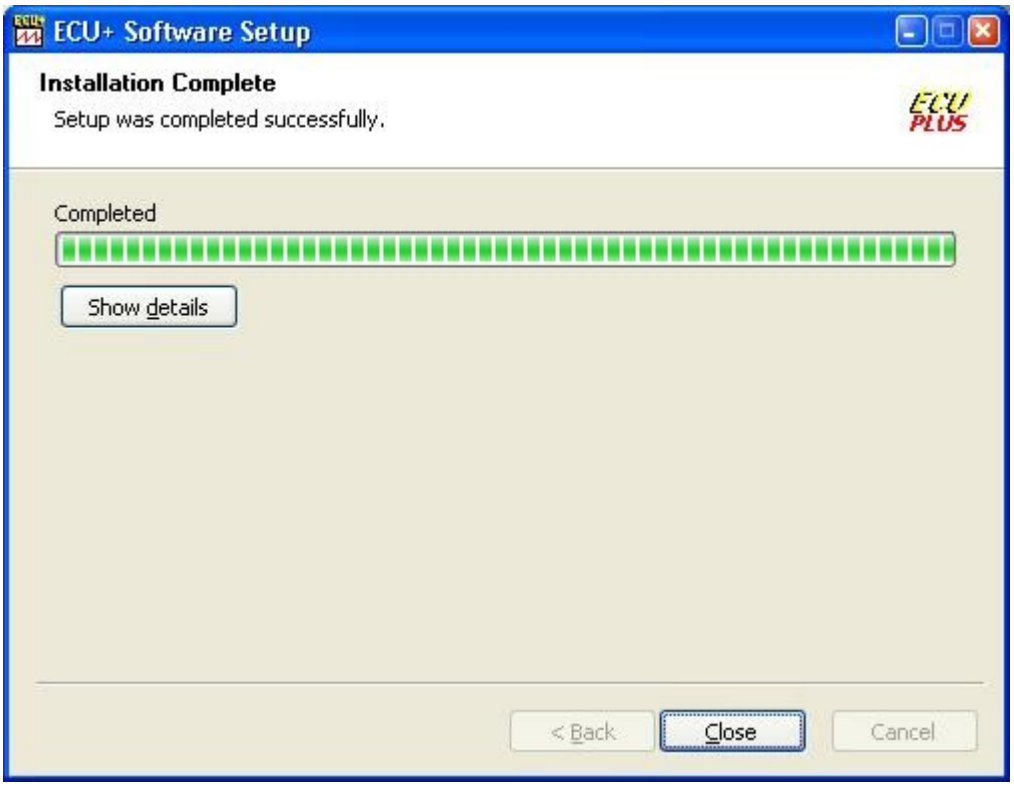


Illustration 13 - The third installation screen

With the software installed, you can start the Windows software by double-clicking the ECU+ Win icon on your desktop.

7 Using the ECU+ Win Software

7.1 Introduction

The ECU+ Win software interfaces to your ECU+ head unit and allows you to tune the ECU+, perform a capture, or just monitor your engine's sensors. The ECU+ Win software also includes advanced data analysis and display capabilities that allows you to easily and accurately configure the ECU+ head unit to make your car perform at peak levels.

7.2 Starting the ECU+ Win Software

To start the ECU+ Win software, double-click the icon on your desktop.

Illustration 14- The ECU+ Win icon

The ECU+ Win software should start up and display its main screen (see Illustration 15).

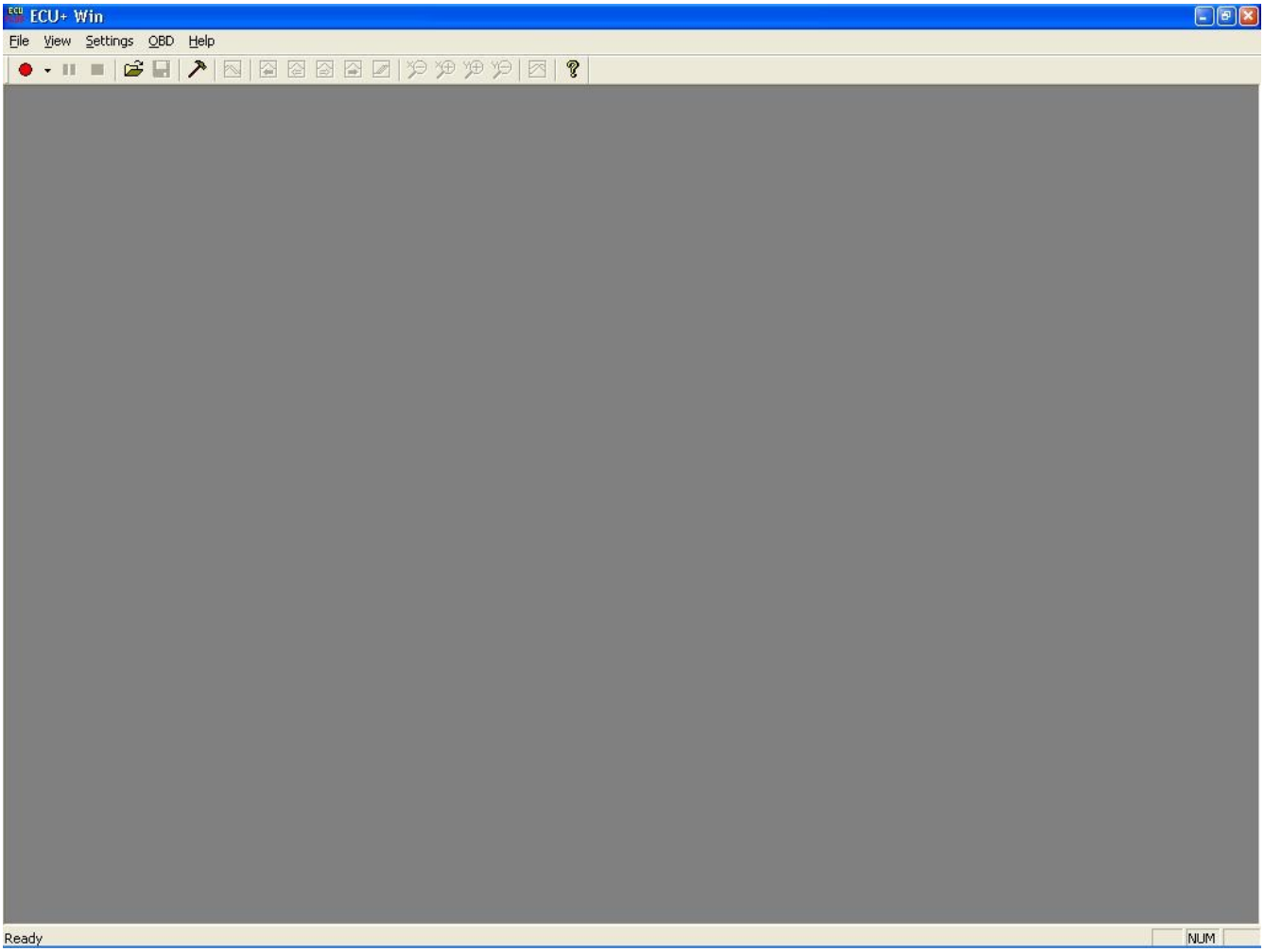


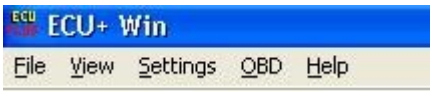
Illustration 15- The ECU+ Win main screen

Because of the amount of information that the ECU+ Win software can display, you might want to maximize the application.

7.3 A Guided Tour of the Software

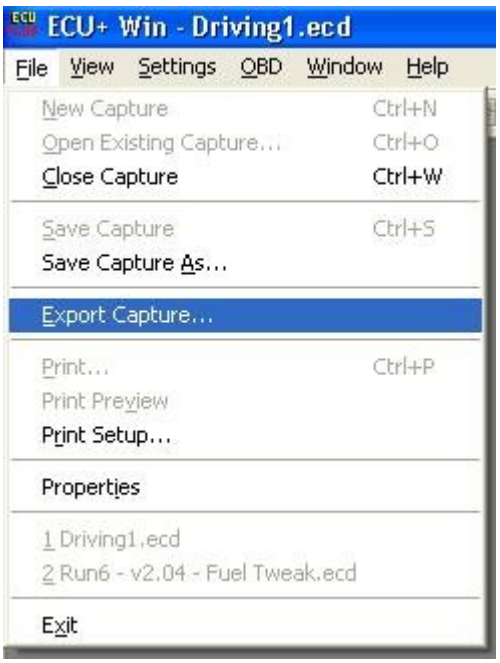
The ECU+ Win application is a relatively complex piece of software. Let's take a look at it one section at a time.

7.3.1 The Menu Bar



The menu bar (see Illustration 16) is at the top of the screen, and includes the usual Windows *File*, *View*, *Help* and other menu items. The menu bar will change slightly depending on what you are doing, but will always include *File*, *View* and *Help*. The sub-menus that are available on the menu bar are as follows:

7.3.1.1 The File Menu



The file menu (see Illustration 17) is used to Open, Save, Close or Export a capture file, to start a New capture, or to exit the ECU+ Win application. You can also print the contents of a graph (discussed later in this manual) from the file menu. Depending upon what you're doing with the ECU+ Win software, one or more of the items on the file menu may be grayed out or not available.

The items on the file menu are as follows:

- *New Capture* - This connects to the ECU+ head unit and begins capturing data from it while displaying the data in real time. Use this menu item to capture new data from your car.
- *Open Existing Capture* - This opens up an existing file on disk containing captured data, and displays the data for viewing or analysis. Use this menu item to display previously-captured data.
- *Close Capture* - When doing a new capture, this finishes the capture and prompts to save the data to disk. When viewing an existing capture, this closes the capture file and prompts to save the data if it's changed. Note that the ECU+ Win software includes an option to automatically name capture files. If this option is enabled, the *Close Capture* command will save new captures automatically without prompting for a filename.
- *Save Capture* - This saves any changes you've made to an open capture file, but doesn't close the file.
- *Export Capture* - This allows you to export the currently open capture to a new file, in comma-separated value (CSV) format. You'll be prompted for the name of the new file. CSV is a common format that can be read by many applications. Most often, you'll use a spreadsheet program (like Microsoft Excel) to read a CSV file for complex analysis and processing.
- *Print* – This prints the current graph view.
- *Print Preview* – Previews the graph view before printing.
- *Print Setup* – Lets you to configure your printer margins and other settings.
- *Properties* – Displays the properties of the current file, including its file name, amount of data and the head unit settings in effect when this capture was started. A sample Properties dialog is

shown in Illustration 18 below.

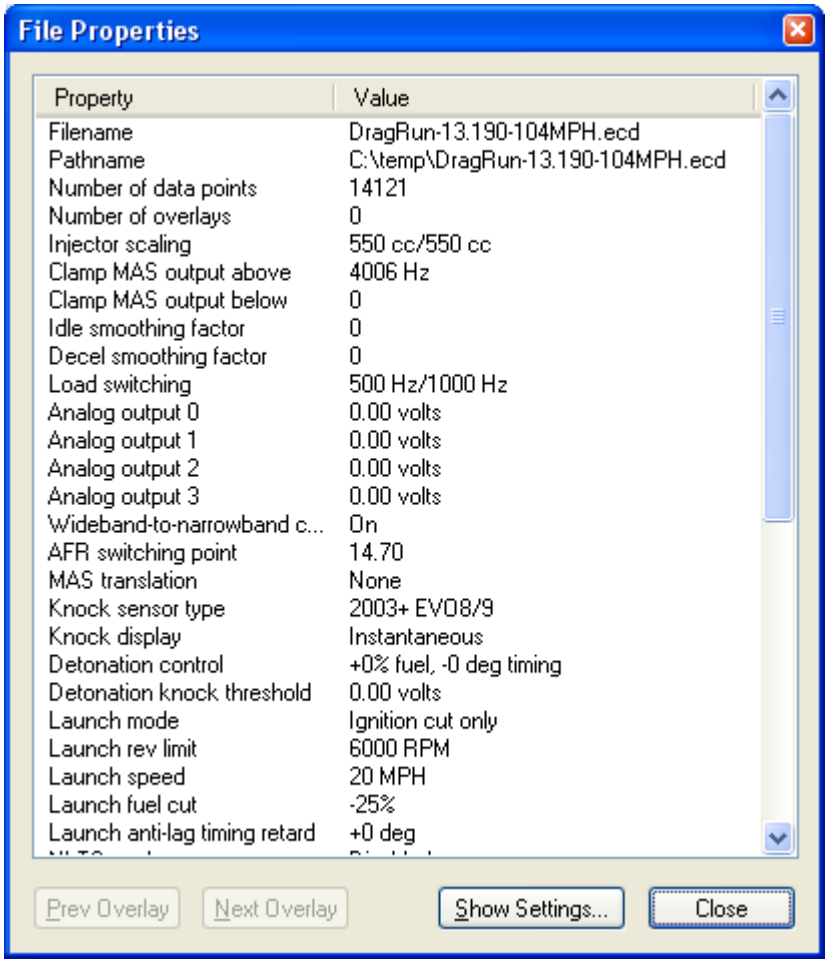


Illustration 18- The file properties dialog

If there are more than one file overlaid onto the primary file, the *Prev Overlay* and *Next Overlay* buttons become active and allow you to see the properties of those files. The *Show Settings* button displays the head unit settings in a separate dialog box.

- Numbered files – The ECU+ Win software remembers the names of several recent capture files that you've opened, and displays them with a number next to them. Selecting a numbered file opens that file quickly, without going through the *Open Existing Capture* menu item.
- *Exit* – This exits the ECU+ Win application, and prompts to save any unsaved capture files.

7.3.1.2 The View Menu

Illustration 19- The view menu

The view menu (see Illustration 19) allows you to toggle on and off the toolbar and status bars for more

on-screen real estate. Simply select a menu item to toggle it on and off.

7.3.1.3 The Window Menu

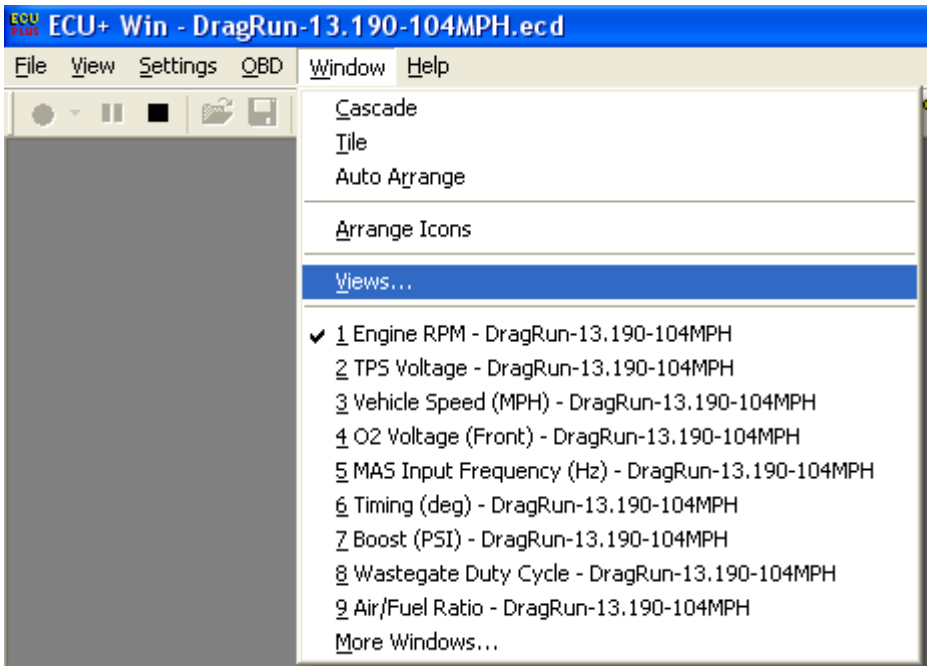


Illustration 20- The window menu

The window menu (see Illustration 20) allows you to arrange the windows in the display area. *Cascade*, *Tile* and *Arrange Icons* work as in other Windows software, but *Auto Arrange* is unique, tiling the visible view windows in a sensible manner. Finally, *Views* shows the available views and allows you to add or remove views from the screen.

Selecting a numbered window on this menu makes the selected window current (in front) and un-minimizes it if necessary. The *More Windows* dialog does the same thing, but is used when more than 9 windows are shown in the display area.

Note that the *Window* menu will only be available when displaying an existing capture file or capturing data to a new file.

7.3.1.4 The Settings Menu

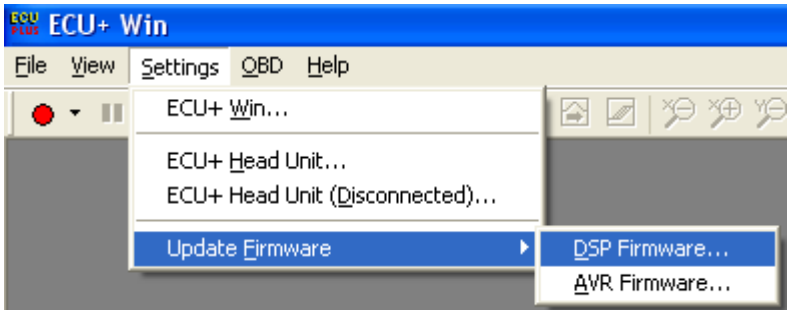


Illustration 21 - The settings menu

The *Settings* menu (see Illustration 21) allows you to configure the ECU+ Win software as well as the ECU+ head unit in either connected or disconnected mode. Select either from the menu, and then

follow the instructions below for setting each up. Additionally, this menu includes options for updating both the DSP firmware and the AVR firmware inside the ECU+ head unit. Updating the firmware is discussed in section Upgrading Your Head Unit Firmware on page 127.

7.3.1.5 The OBD Menu

Illustration 22 - The OBD menu

The *OBD* menu (see Illustration 22) invokes the OBD functionality of the ECU+. This functionality will be discussed in the section On Board Diagnostics (OBD) on page 108.

7.3.1.6 The Help Menu

The *Help* menu has a single menu item which displays the *About* dialog for the ECU+ Win application. The about dialog just displays the software version and copyright information.

7.3.2 The Toolbar

Illustration 23- The toolbar

The toolbar (see Illustration 23) is a quick way to access various ECU+ Win menu items with just a single click. The items on the toolbar, and the menu item equivalences are (from left to right):

- *New Capture* - The record icon. This begins capturing data from the ECU+ head unit. This is equivalent to *File->New Capture*.
- *Auto-Name Capture Drop-Down Menu* – The down-arrow next to the record icon. This invokes a drop-down menu that toggles on and off the auto file naming option.

Illustration 24 - The auto-name drop-down menu

- *Pause Capture* - The pause icon. This button temporarily pauses the current capture so that you can scroll back and look at old data. Click this button once to pause the capture, and then again to restart the capture. Any data received from the ECU+ head unit while the capture is paused is lost. There is no menu item equivalent to this button.
- *Close Capture* - The stop icon. This closes a new capture or existing capture file, and prompts

to save the data if it's changed. This is equivalent to *File->Close Capture*.

- *Open Existing Capture* - The open folder icon. This opens an existing capture file for analysis. This is equivalent to *File->Open Existing Capture*.
- *Save Capture* - The floppy disk icon. This manually saves the current capture if it's changed. This is equivalent to *File->Save Capture*.
- *Setup Head Unit* - The hammer icon. This brings up the dialog box to configure the ECU+ head unit's fuel and timing maps and other parameters. This is equivalent to *Settings->ECU+ Head Unit*.
- *Add Overlay* - The two-graph icon next to the hammer icon. This adds a new overlay to the current capture file for comparison purposes. This is equivalent to *Graph Context Menu->Overlaid Plots->Add New*.
- Overlay Shifting - The next four icons (which show a graph and an arrow) are shortcuts used for shifting the most recent overlay to the left or right. The icons represent shifting left 1/4 page, left by one, shift right by one, and right by 1/4 page, respectively. These are equivalent to *Graph Context Menu->Overlaid Plots->Shift Last*.
- *Dyno Align All* – This aligns all of the overlays to the same point in preparation for a dyno analysis. This is is equivalent to *Graph Context Menu->Overlaid Plots->Shift Last*.
- Graph Zooming - The next four icons (magnifying glass-like) are used to zoom a graph's X or Y axis in or out, and are equivalent to *Graph Context Menu->Zoom X(Y) Axis->In(Out)*. They zoom the current graph out on the X axis, in on the X axis, out on the Y axis and in on the Y axis, respectively.
- *Dyno Analysis* - The dyno icon. This brings up the dyno analysis dialog. This is equivalent to *Graph Context Menu->Analyze->Dyno Analysis*.
- *Help* - The question mark icon. Displays the current ECU+ Win “About" dialog. This is equivalent to *Help->About ECU+ Win*.

Note that the toolbar can be toggled on and off with the View->Toolbar menu item. If you never use the toolbar, you may want to turn it off to save screen real-estate.

7.3.3 The Status Bar

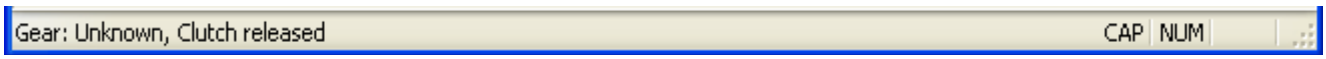


Illustration 25- The status bar

The status bar (see Illustration 25) is used to display the settings of the keyboard caps-lock (CAP), num-lock (NUM) and scroll lock (SCRL) keys, as well as longer descriptions of menu items. Additionally, if a capture is open, the status line will display some details about the currently-highlighted data point, like the clutch position and the gear. The status line can be turned off with the *View->Status Bar* menu item.

7.3.4 The Display Area

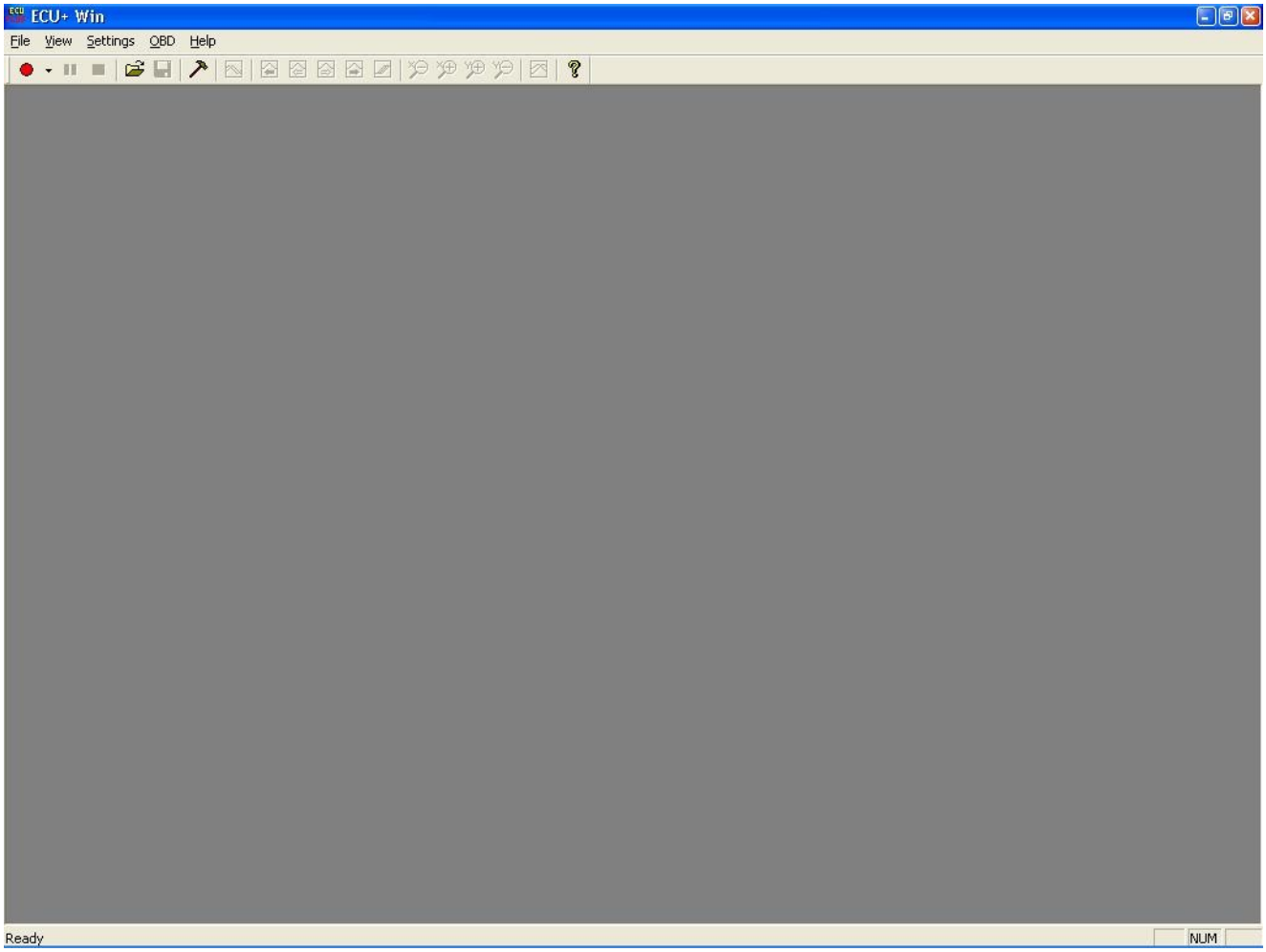


Illustration 26- The display area

The display area makes up the bulk of the ECU+ Win's screen space. Inside the display area, the software will display “views” of a live or captured data file. A view is just a sub-window in which data is displayed. Views will either be text boxes, containing data values numerically, or graphs showing data in graph form. More information on views is available in section Setting Up Views on page 65.

7.4 Creating a New Capture

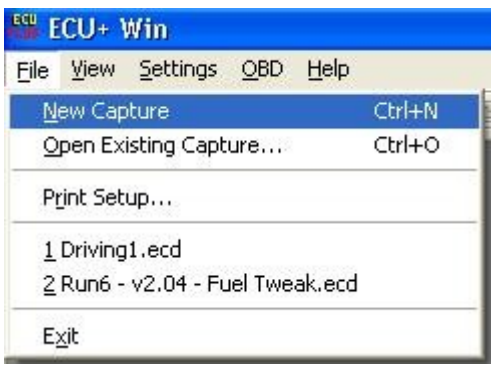


Illustration 27 - The new capture menu item

The ECU+ Win software can capture and store away "datalogging" information from the ECU+ head unit. These captures grab various engine sensor values 25 times per second and store them away in a

file on your computer's hard disk. This captured data contains a permanent record of your engine's operation, and can be later loaded back into the ECU+ Win software for analysis or comparison to other captures.

To create a new capture file, use the *File->New Capture* menu item, or click on the record icon in the toolbar. The ECU+ Win software will then open up the serial port and try to connect to the ECU+ head unit. Once the connection is established, the software will open up the views that you've configured and begin capturing data from the head unit. While the capture is going on, you can open, close or resize any of the views in the display area. At any point during the capture, you can use File->Close or click the stop icon in the toolbar to stop capturing. When you stop capturing, ECU+ Win will prompt for you to save the file, and display the standard Windows file-save dialog. At any point during the capture, you can also pause the capture by clicking the pause icon in the toolbar. Any data accumulated from the head unit while paused will be lost.

The ECU+ Win software can auto-name your capture files, or prompt you for a filename. To select the desired option, click the arrow to the right of the record icon. To configure how auto file naming works, see the section Auto File Naming Configuration on page 43.

7.5 Opening an Existing Capture File

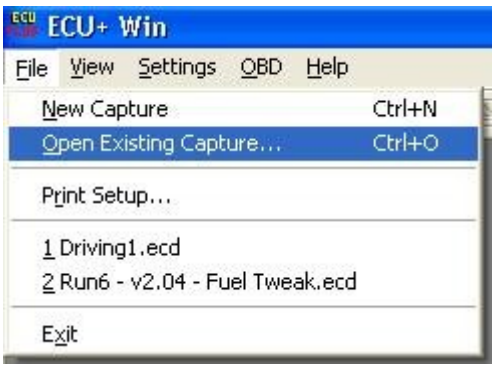


Illustration 29 - The open existing capture menu item

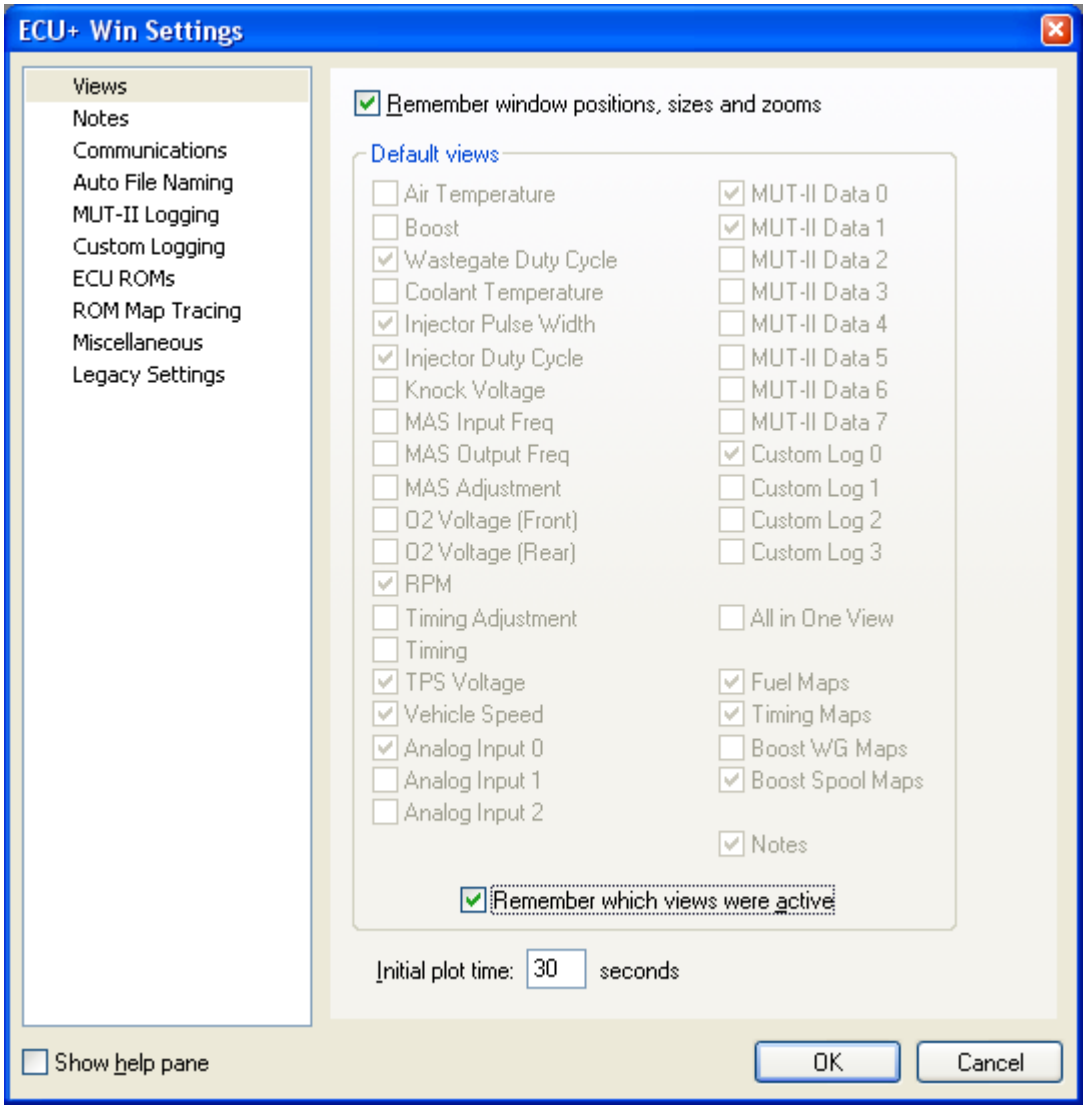
Captures that you've acquired with the ECU+ Win software can be loaded into ECU+ Win by "opening" an existing capture file. To do this, use either the *File->Open Existing Capture* menu item, or click the open folder icon on the toolbar and select the capture file you'd like to view. ECU+ Win will open this file and display its contents within the views you've configured. When you're done with this capture, use *File->Close* capture to close it out.

Note: Only one capture file can be open at a time. You can, however, overlay one capture on top of another. See the section Overlaying Multiple Capture Files on page 66 to see how this powerful technique works.

7.6 Configuring the ECU+ Win Software

To configure the ECU+ Win software, use the *Settings->ECU+ Win* menu item. This opens the ECU+ Win configuration dialog. This has five tabs:

7.6.1 View Configuration

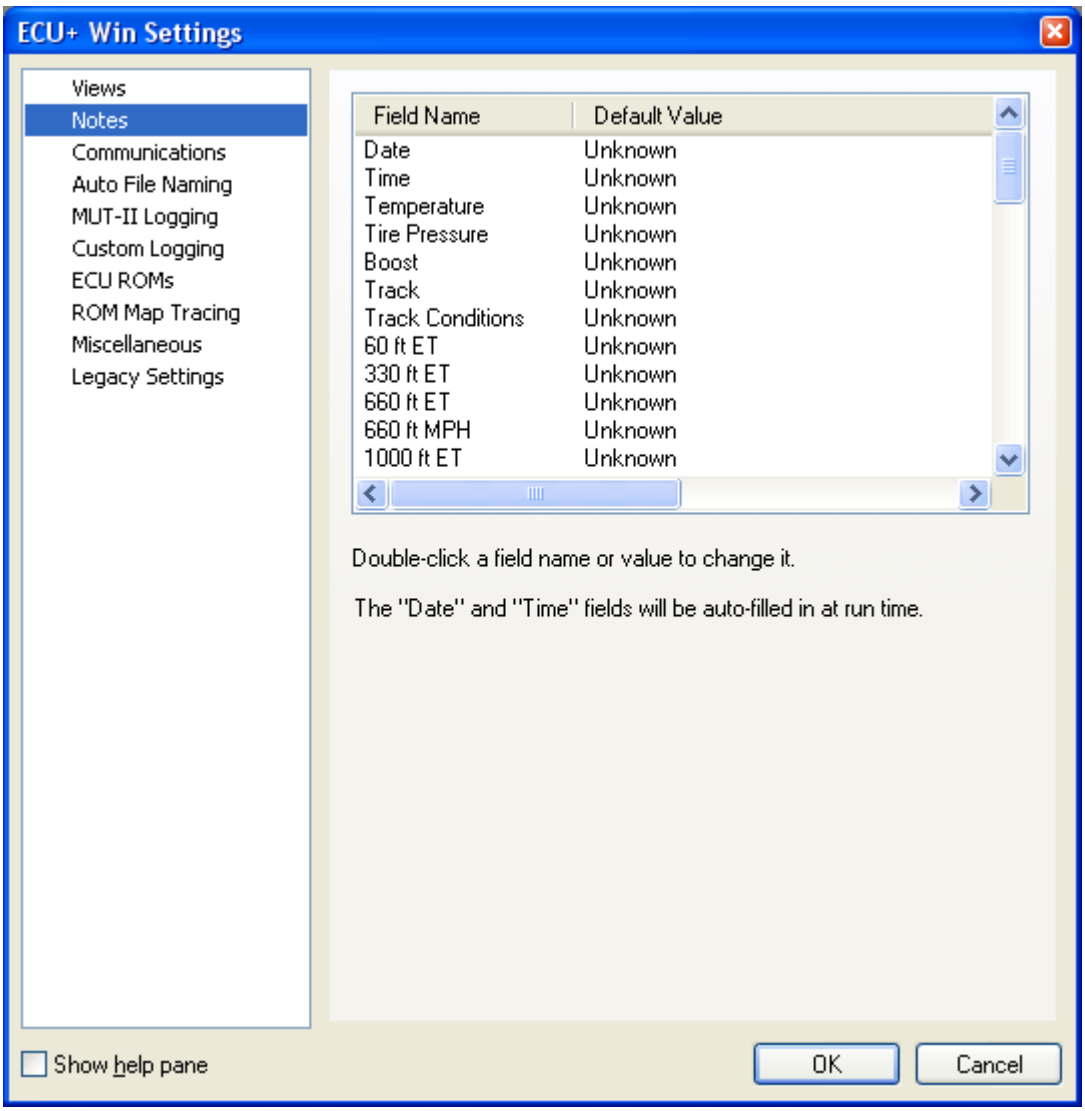


The view configuration dialog tells the ECU+ Win software what views you'd like to see by default. When you start a new capture or open an existing capture file, the ECU+ Win software starts up and displays one or more views of the data associated with this capture. If *Remember which views were active* is checked, the ECU+ Win software will display the same set of views that were active the last time you showed a capture. If this is un-checked, the checked views will be shown instead,

The checkbox *Remember window positions, sizes and zooms* determines whether the ECU+ Win software will restore the views in the same window position, minimized status, and overlap as before.

The *Initial plot time* text field sets how many seconds the X axis of the graph views will display by default.

7.6.2 Notes Configuration



The Notes configuration dialog allows you to configure which notes get added to new captures by default. When you start a new capture, the ECU+ Win software automatically copies the values defined here into the notes view for that capture. You can add or delete values from this dialog to configure what notes are used by default. If a field named “Date” and/or a field named “Time” is defined here, it's filled in by the ECU+ Win software when the new capture is started.

Double-click a field name or value to change it, and then press ENTER.

7.6.3 Communications Configuration

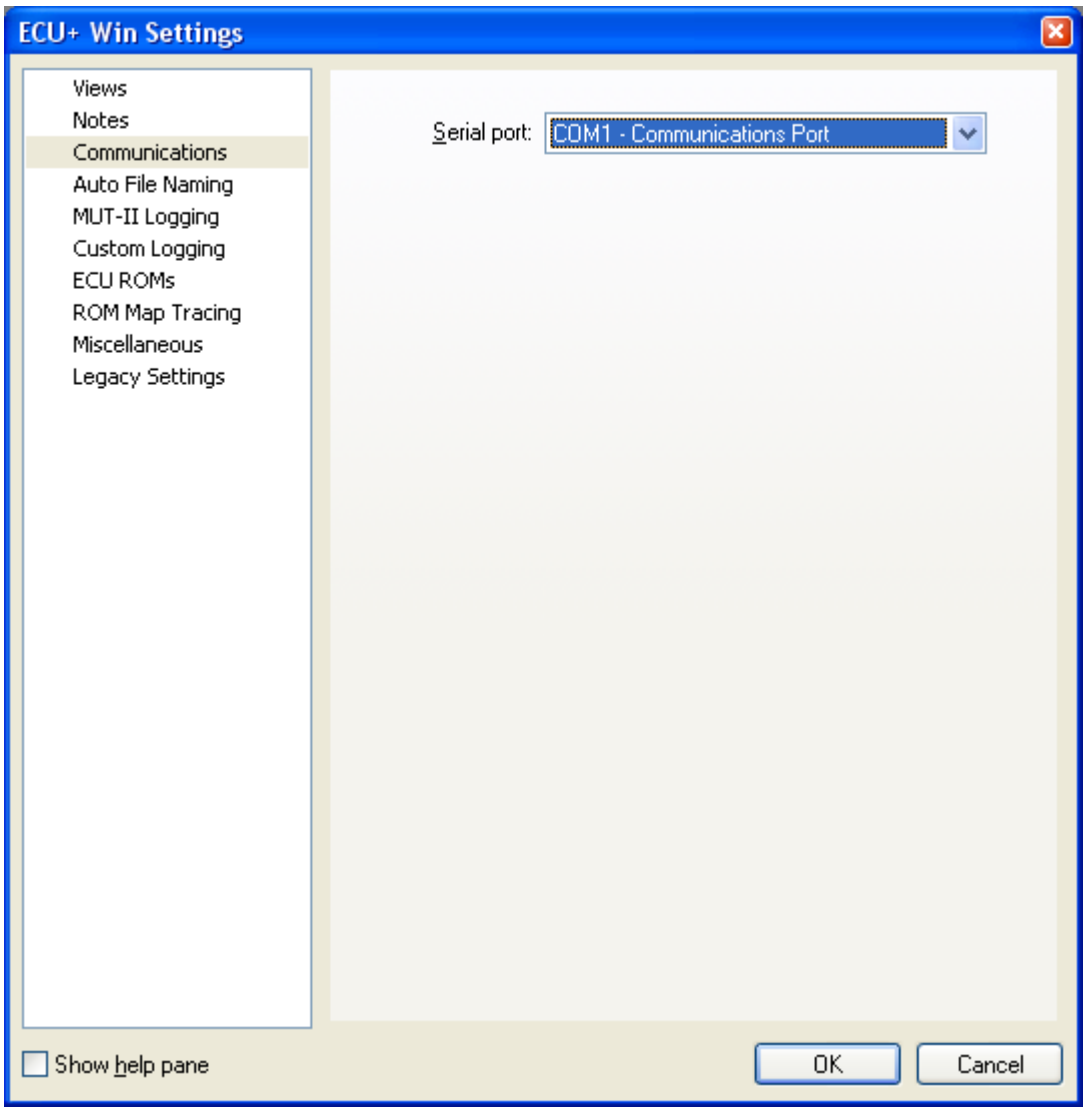


Illustration 32 - The communications configuration dialog

The communications configuration dialog sets up what serial port your laptop will use when talking to the ECU+ head unit. Typical values used are COM1 (with a laptop that has a "real" serial port), or COM5 (with a laptop using a USB-to-serial adapter). Click the dropdown arrow to change this value.

7.6.4 Auto File Naming Configuration

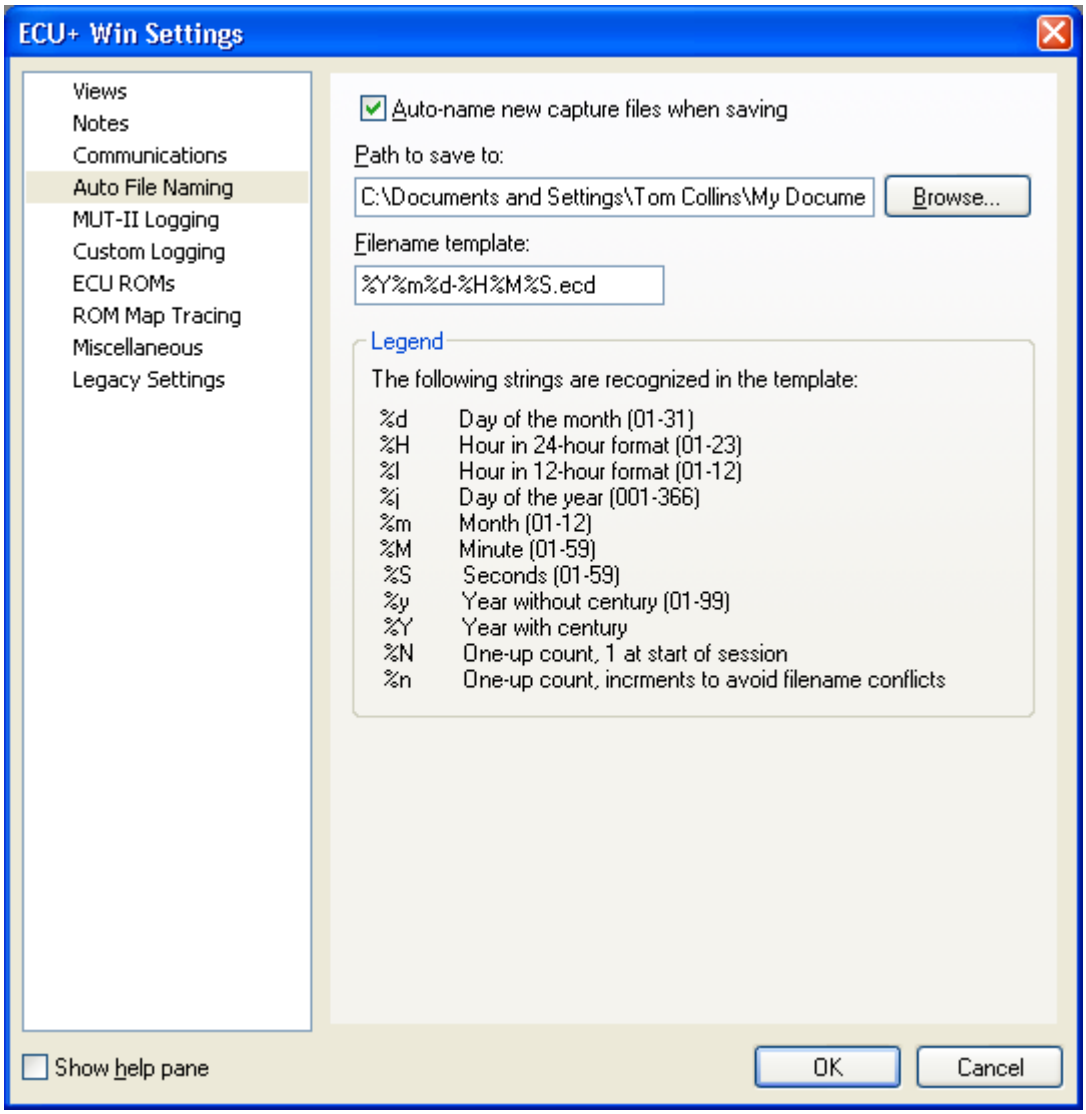


Illustration 33 - The auto file naming configuration dialog

The auto file naming configuration dialog tells the ECU+ Win software how you'd like auto file naming to work. When you start a new capture with auto file naming on, these settings determine where the capture file is saved to, and what its name is. Use the *Browse...* button to select a directory to place the captures in, and enter a filename template for the filename. The template automatically replaces several "magic strings" in the filename as shown in the legend to name your capture files according to the date of capture or a one-up count.

7.6.5 Custom Logging Configuration

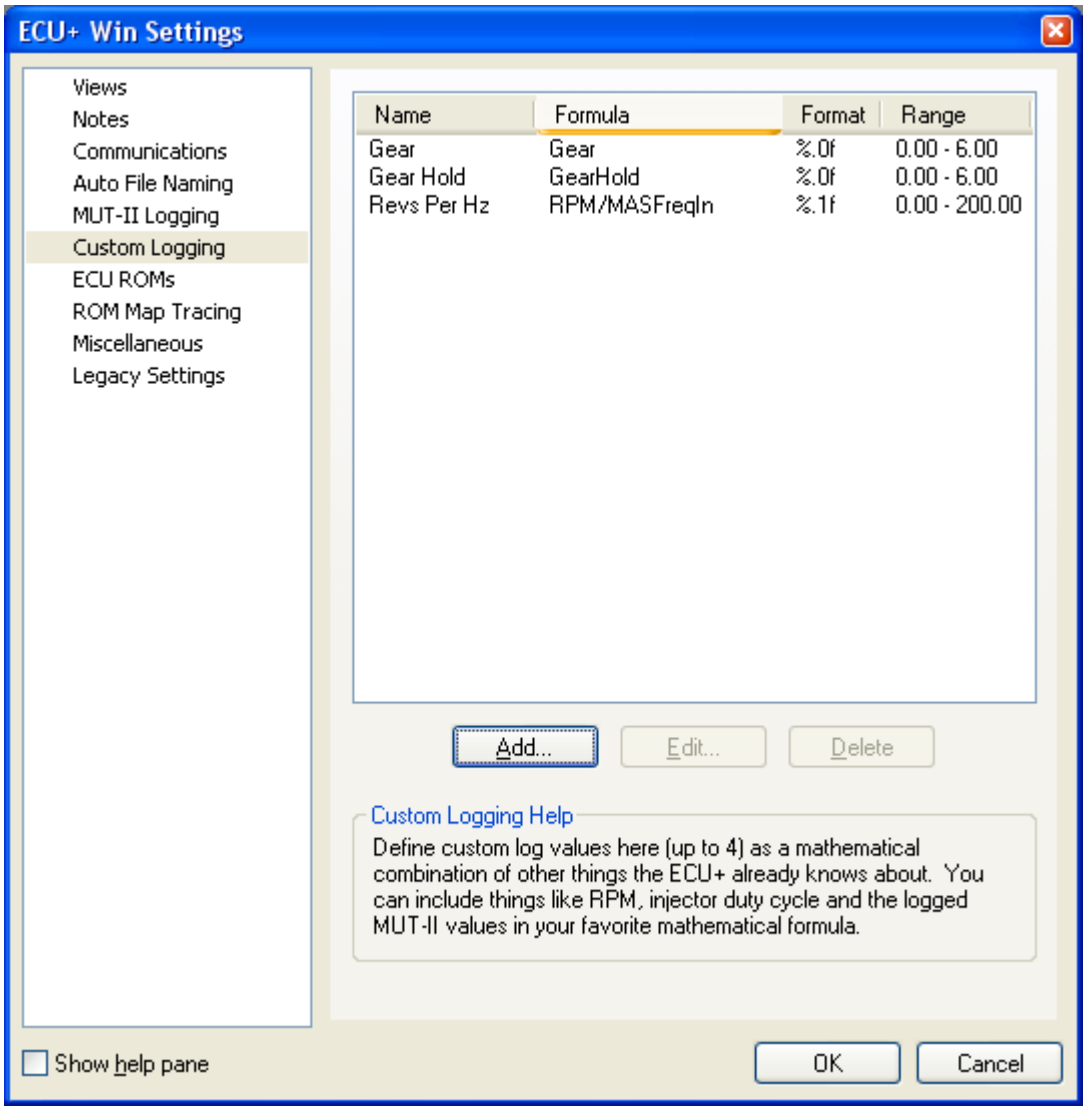
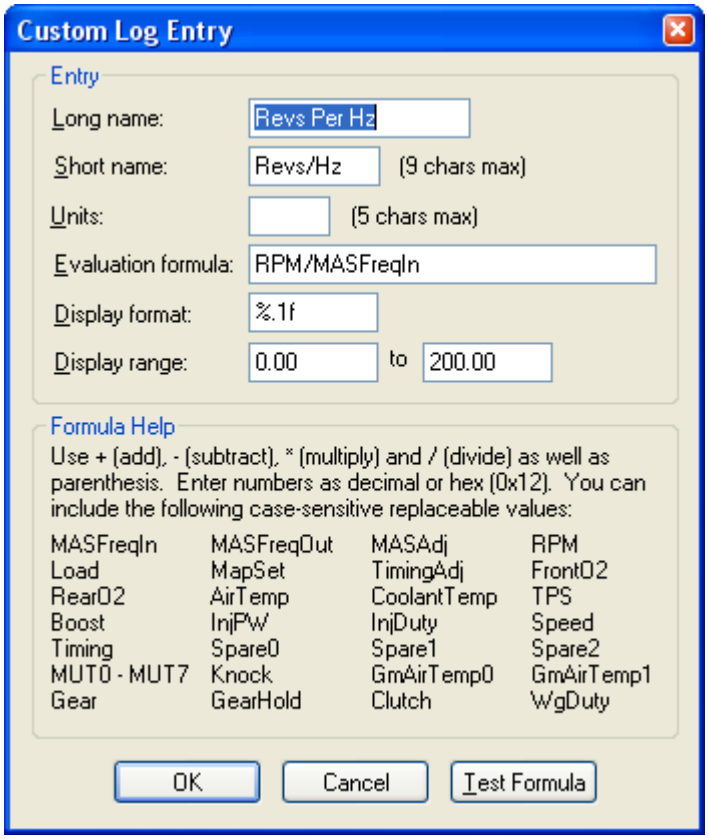


Illustration 34 - The custom logging configuration dialog

The custom logging configuration dialog allows the ECU+ Win software to datalog up to four "custom" values. Custom values are created by applying a mathematical formula to values that the ECU+ already knows about.

On this page, the *Add*, *Edit* and *Delete* buttons are used to manipulate the list of custom values. Adding or editing a single custom value invokes this dialog:



To configure a custom log entry, there are a number of parameters that need to be specified.

- The *Long name* and *Short name* fields specify the name to be used for this custom value in various parts of the ECU+ Win software.
- The *Units* field specifies what units should be displayed alongside the custom value. Typical values for this field are Hz for frequencies or % for percentages.
- The *Evaluation formula* determines the mathematical formula that the ECU+ Win software should use to create this custom value. Parenthesis, constant values (in decimal or hex, if preceded by 0x), and the add, subtract, multiply and divide operators can be used here along with a number of replaceable strings (as listed in the Formula Help) display. A simple example of an evaluation formula is RPM*2. This will datalog the RPM in a separate window. Changing the formula to RPM*2+500 also works as expected, as does RPM/MASFreqIn. Use the *Test Formula* button to see if a complex formula evaluates as you expect.

  Most values allowed in the Evaluation formula are self-explanatory, with the following exceptions:

String	***Interpreted As***
Load	0 for idle, 1 for low load, 2 for medium load, 3 for high load, 4 for WOT.
MapSet	0 for the primary map set, 1 for the secondary map set.
TPS	The TPS voltage, 0-5. Unaffected by the option to show TPS as a percentage.
Spare 0/1/2	The voltage on the spare inputs, 0-5. Spare 1 and

String	*Interpreted As*
	Spare 2 can also be used as inputs for EGT or wideband O2 kits, but the custom value for these is always the raw voltage.
MUT0 through MUT7	These are the raw MUT-II byte (value: 0-255) as read from the stock ECU.
MASFreqIn	The frequency of the MAS input waveform. When the ECU+ is converting the waveform of a GM MAF (which reads very high frequencies), this will be the post-translation frequency.
Clutch	0 when the clutch is release, 1 when the clutch is pressed in. This can only be logged through the custom logging facility.
Gear	The ECU+'s calculated gear number, 1-6. If the ECU+'s calculation can't figure out what gear the car is in (bad gear ratios, vehicle stopped or at very low speeds), or if the clutch is pressed in, this value will be 0. This can only be logged through the custom logging facility.
GearHold	As above, 1-6, but "holds" the previous gear instead of changing to 0 during unusual time. This value never goes to zero. This can only be logged through the custom logging facility.
Boost	The voltage on the MAP sensor input, 0-5, rather than the boost level in PSI or bar.
AirTemp and CoolantTemp	The air and coolant temperatures in degrees F. If the International version of the software is installed, these hold the temperature in degrees C.
GmAirTemp0 and GmAirTemp1	The temperature as read from the two ECU+ GM air intake temperature sensor inputs. As with AirTemp, these are in deg F or deg C depending on the software version. These two can only be logged through the custom logging facility.

- The *Display range* field specifies the lower and upper values expected from the result of the entered evaluation formula. This is used to set the range limits on the graph views displayed in the software.
- The *Display format* field specifies the precision used to display the calculated value. Assuming the calculated value is 100, some examples are:

Display Format	*Shows As*
%.0f	100
%.1f	100.2
%.2f	100.25

7.6.6 MUT-II Logging Configuration

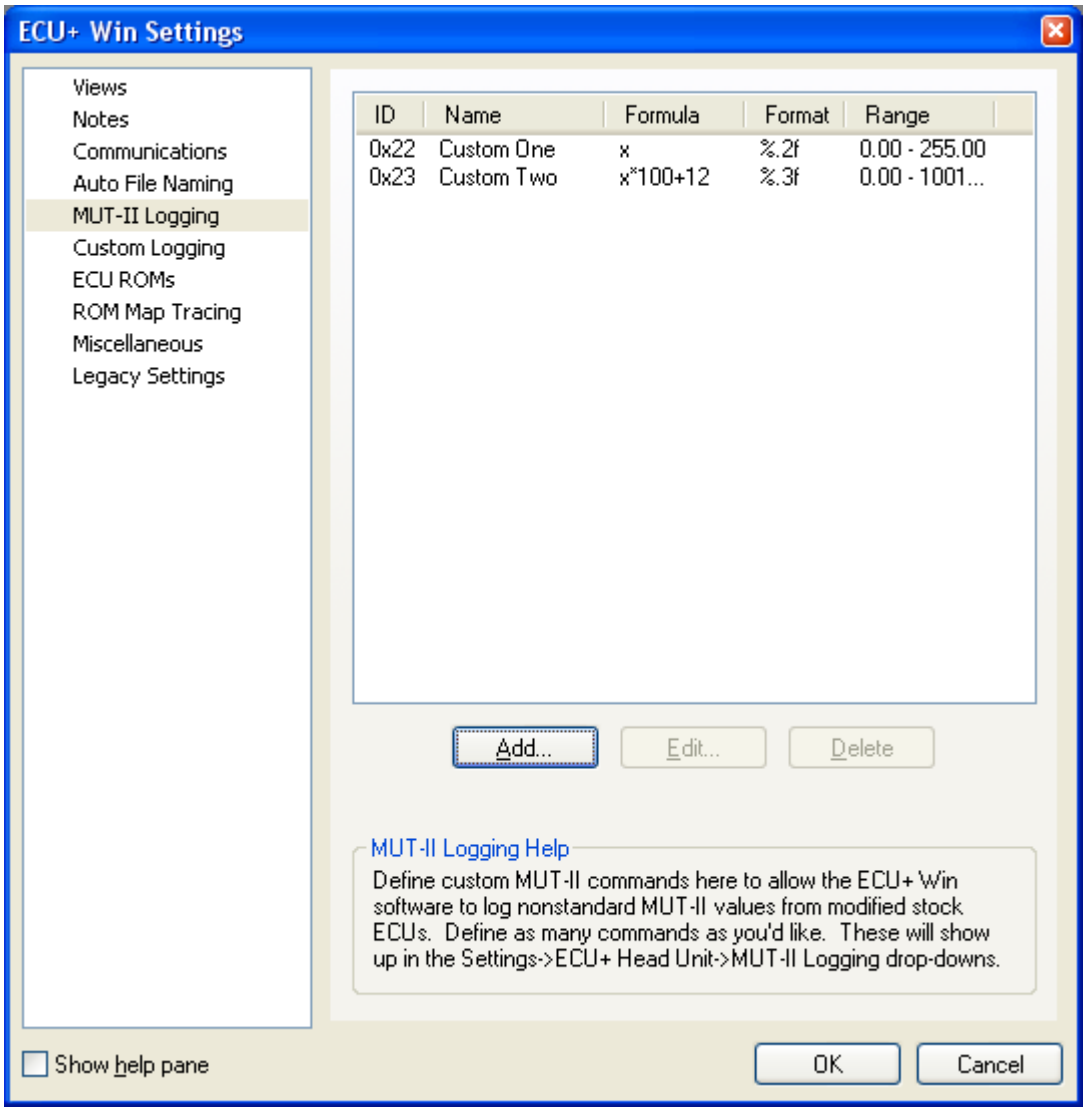


Illustration 36 - The MUT-II logging configuration dialog

The ECU+ can log data values from the stock ECU using a protocol called MUT-II.

The MUT-II protocol sends various command bytes to the stock ECU over the OBD-II port, and reads the single-byte response code. By default, the ECU+ knows how to interpret a basic set of MUT-II commands. The MUT-II logging configuration dialog allows you to define additional custom MUT-II commands for modified stock ECU's or to extend the ECU+'s functionality for non-standard MUT-II commands. Custom MUT-II commands will be shown in the *Settings->ECU+ Head Unit->MUT-II Logging* drop-down list.

On this page, the *Add*, *Edit* and *Delete* buttons are used to manipulate the list of custom MUT-II commands that the ECU+ head unit understands. Adding or editing a single MUT-II command invokes this dialog:

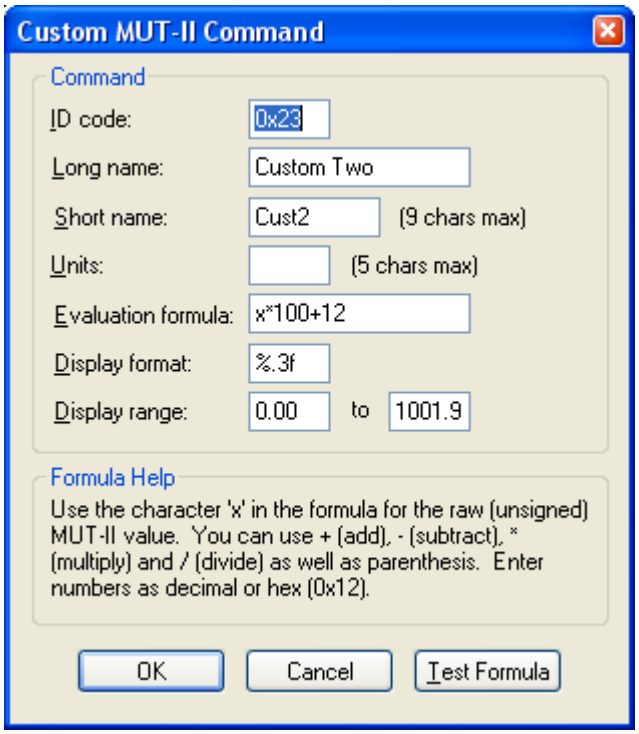


Illustration 37 - The custom MUT-II command editing dialog

To configure a custom MUT-II command, there are a number of parameters that need to be specified.

- *ID code* is the command number, specified as a regular decimal number or a hex number (for example, 0x10 for the value 16). This is the byte that is sent to the stock ECU.
- The *Long name* and *Short name* fields specify the name to be used for this MUT-II command in various parts of the ECU+ Win software.
- The *Units* field specifies what units should be displayed alongside the custom MUT-II value. Typical values for this field are Hz for frequencies or % for percentages.
- The *Evaluation formula* determines how the ECU+ Win software should interpret the response byte returned by the stock ECU. The response comes back as a byte called x, and that can be manipulated with the evaluation formula. As an example, to double the returned value before displaying it in ECU+ Win, simply enter x*2 in the *Evaluation formula* field. Decimal or floating-point values can be used in the Evaluation formula field, as can parenthesis and simple operators like add, subtract, multiply and divide. Use the *Test Formula* button to see if a complex formula evaluates as you expect.
- By default, the display range when graphing the result of evaluating the MUT-II value is 0 to 255 (since that's the numeric range of a single byte). If you've manipulated the value with the Evaluation formula, enter a different range in the *Display range* fields. For example, if the stock ECU returns a byte in the range 0-32, and your evaluation formula is x*3, set the *Display range* to 0 to 100.
- The *Display format* field specifies the precision used to display the calculated value. Assuming the calculated value is 100, some examples are:

Display Format	*Shows As*
%.0f	100
%.1f	100.2
%.2f	100.25

7.6.7 ECU ROMs Configuration

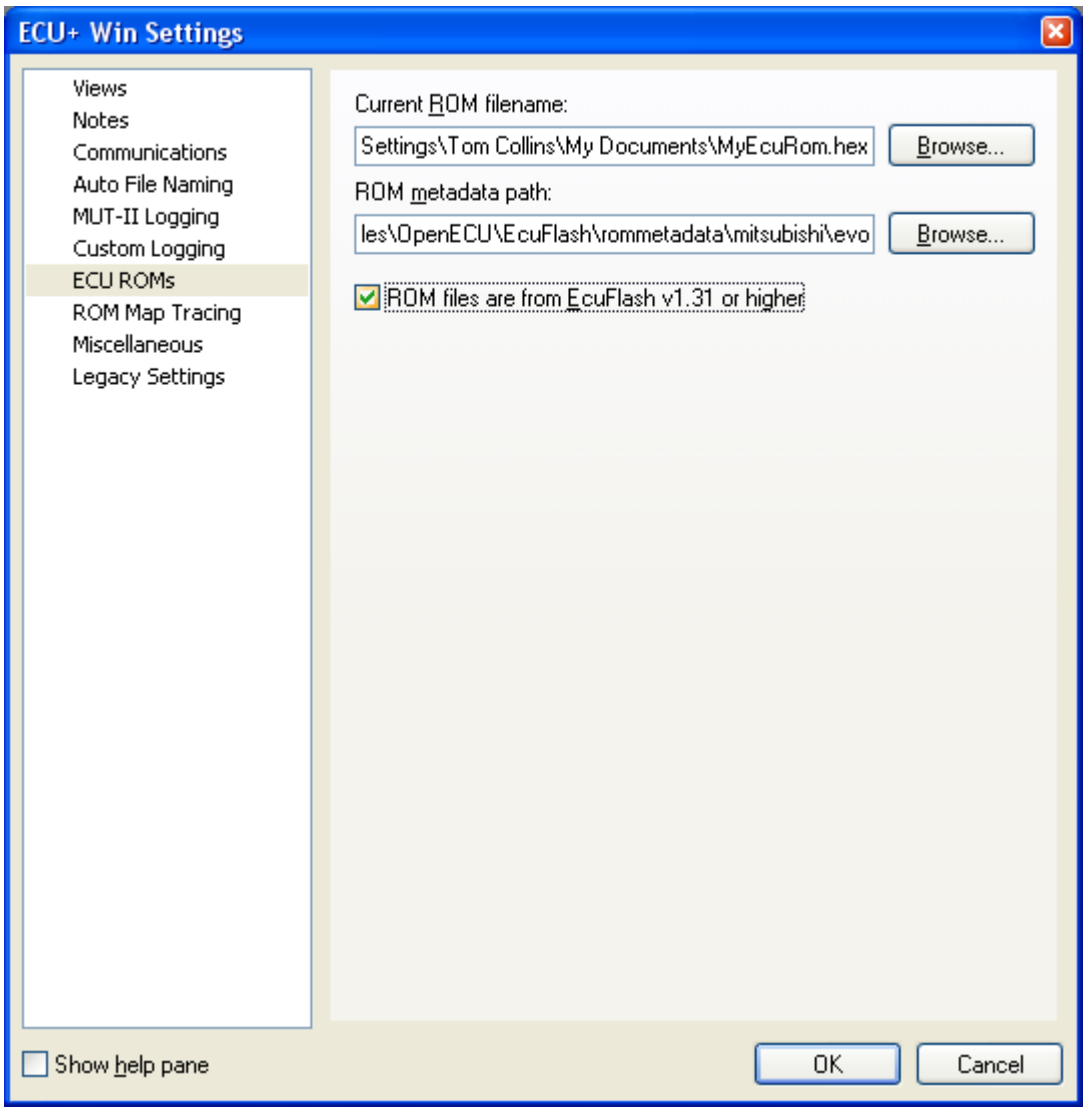


Illustration 38 - The ECU ROMs configuration dialog

The ECU ROMs configuration dialog configures your current ROM file and metadata path. The ROM metadata path is where ECU+ Win looks for the XML files that describe your ROM file from your stock ECU. Typically, this points to the *rommetadata\mitsubish\evo* directory underneath your EcuFlash installation directory. The ROM filename is the actual ROM file downloaded from your stock ECU.

Note that this dialog is not present unless you've installed the version of ECU+ Win that allows ROM editing and map tracing.

7.6.8 ROM Map Tracing Configuration

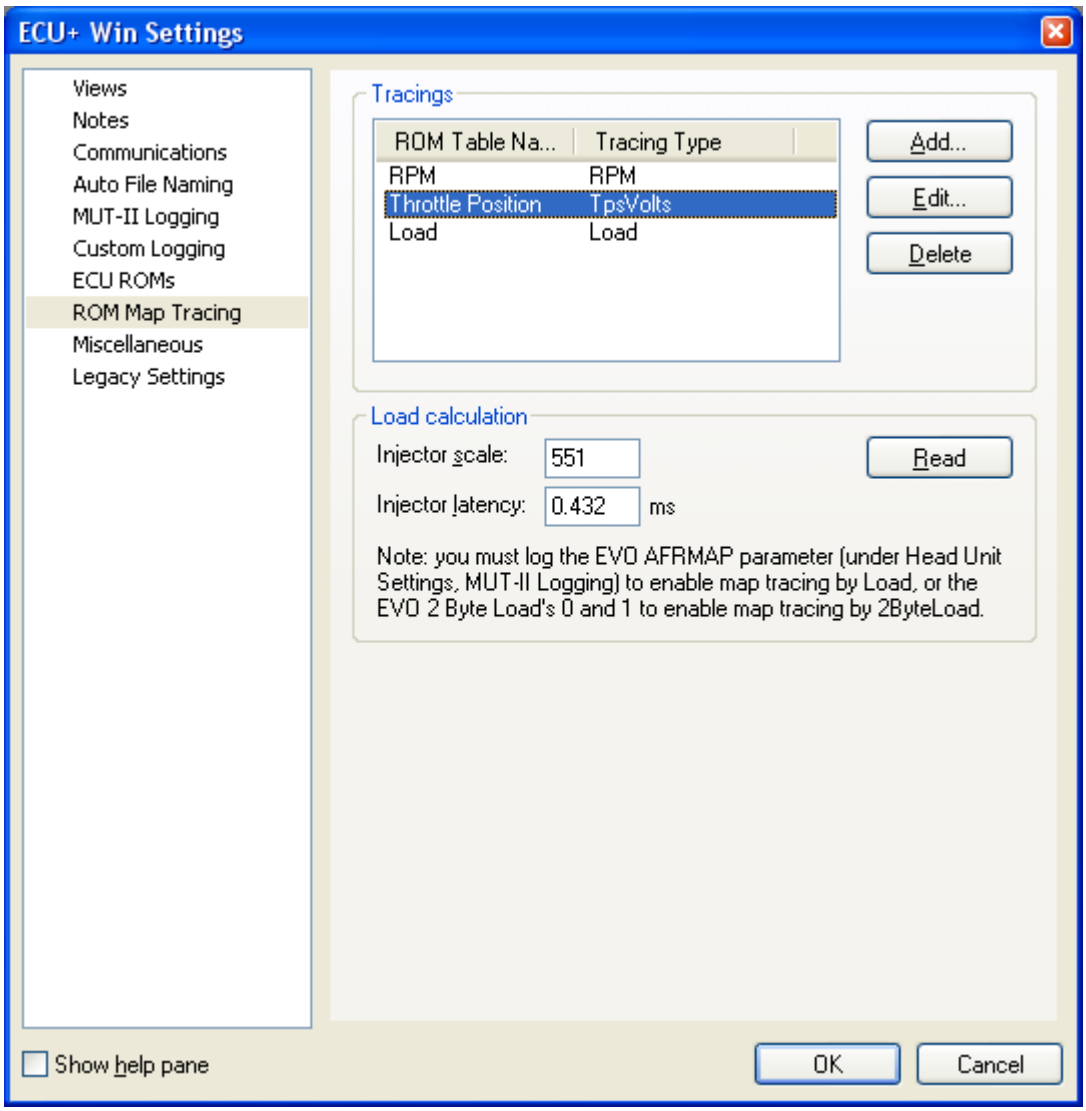


Illustration 39 - The ROM map tracing configuration dialog

The ROM map tracing configuration dialog configures how the ECU+ "traces" the ROM maps extracted from the stock ECU. Map tracing allows the current cell of a given ECU map to highlight as the cursor is moved within the captured ECU+ data stream. With the ECU+'s dual fuel, timing and boost control maps, tracing is automatic – no configuration is necessary. For the stock ECU maps, maps are defined externally through a set of XML files. This dialog associates a value that the ECU+ software knows about (the *Tracing Type*) with the X or Y axis name of a stock ECU ROM (the *ROM Table Name*).

To configure a tracing type, simply add an entry to the *Tracings* list.

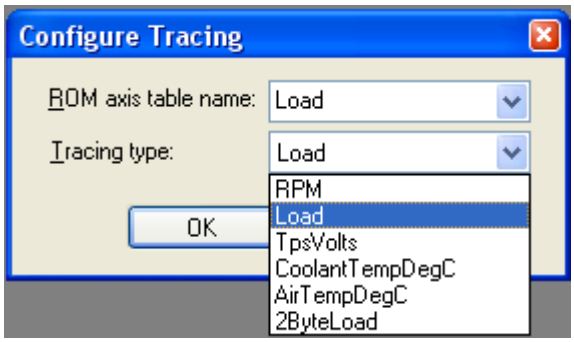


Illustration 40 - The tracing configuration dialog

For each ROM X or Y axis name, select one of the ECU+'s known tracing types and hit Ok. Generally, only a few of the ROM X or Y axis names need to be associated with a tracing type – RPM and Load are the most widely used.

The *Load* tracing type is special. Load is computed inside of the ECU+ Win software, but the calculation needs specific values for the injector scale and injector latency values used inside the stock ECU. Enter new values in the text field, or use the *Read* button to read the values from current ROM file.

Note that this dialog is not present unless you've installed the version of ECU+ Win that allows ROM editing and map tracing.

7.6.9 Miscellaneous Configuration

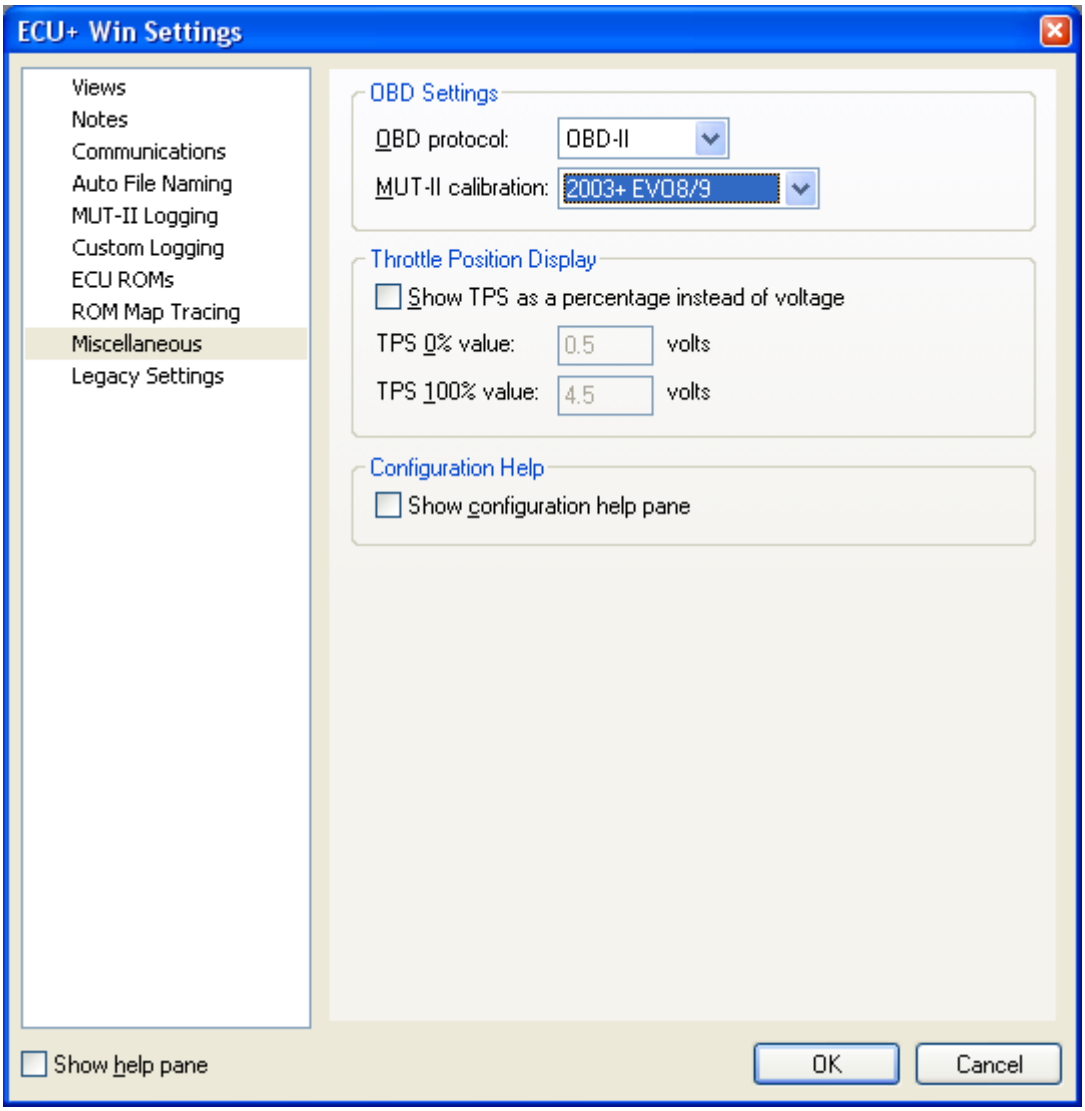


Illustration 41 - The miscellaneous configuration dialog

The miscellaneous configuration dialog lets you configure several miscellaneous parameters:

- *OBD Protocol* – The ECU+ can communicate with the stock ECU to display diagnostic information from the main menu's OBD menu entry. This selects what protocol will be used. Two protocols are available:
  - *MUT-II* – this is protocol is supported on all cars that the ECU+ works with.
  - *OBD-II* – only supported on the 1996 and newer DSMs and the Evolutions. OBD-II displays more information than MUT-II, but is not supported on the earlier cars.

  Click the dropdown for one of these to change the current values..
- *MUT-II Calibration* – When interpreting data in the MUT-II format, ECU+ Win can interpret several of the values in different ways. This selects what interpretation the software will use. Select the calibration that matches your vehicle here.
- *Throttle Position Display* – The ECU+ retrieves the throttle position signal from the stock ECU as a voltage, where 0 volts is throttle completely closed and +5 volts is throttle completely open.

Most cars don't read 0 volts when the throttle is completely closed, and some people prefer to see throttle position shown as a percentage to make TPS easier to interpret. This setting determines how the ECU+ will display TPS. When displaying TPS as a percentage, enter low and high voltages that correspond to 0% and 100% TPS.

- *Configuration Help* – All of the ECU+ configuration screens (the ECU+ Win Settings dialogs as well as the ECU+ Head Unit Settings dialogs) include an optional help pane that will display context-sensitive help for each configuration option. Use the *Show configuration help pane* button to set whether this help is shown by default. Help can always be toggled on and off as needed with the *Show help pane* button in the bottom left of each configuration screen. The miscellaneous configuration dialog with the help pane enabled is show in Illustration 42 below.

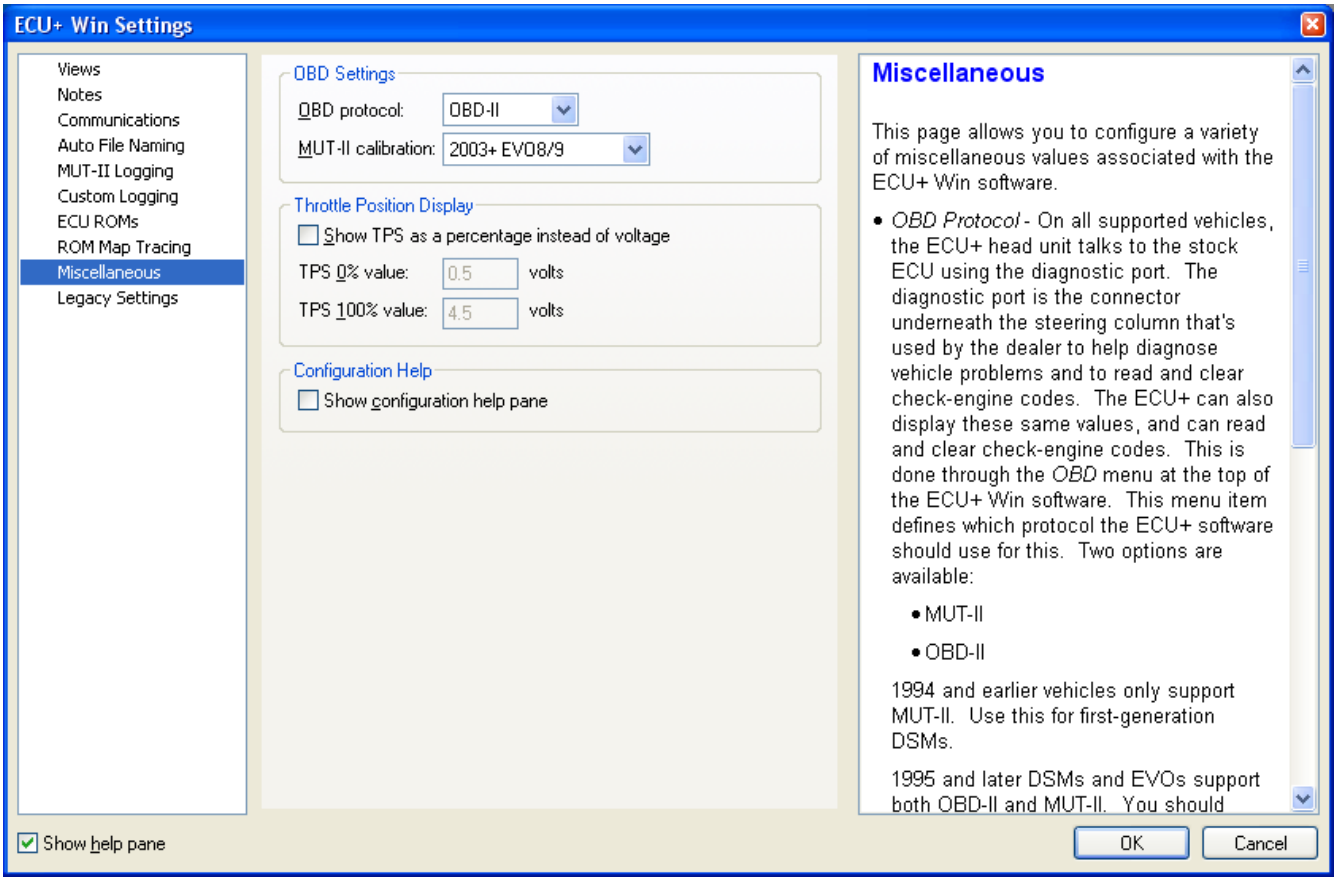


Illustration 42 - A sample help pane

7.6.10 Legacy Settings Configuration

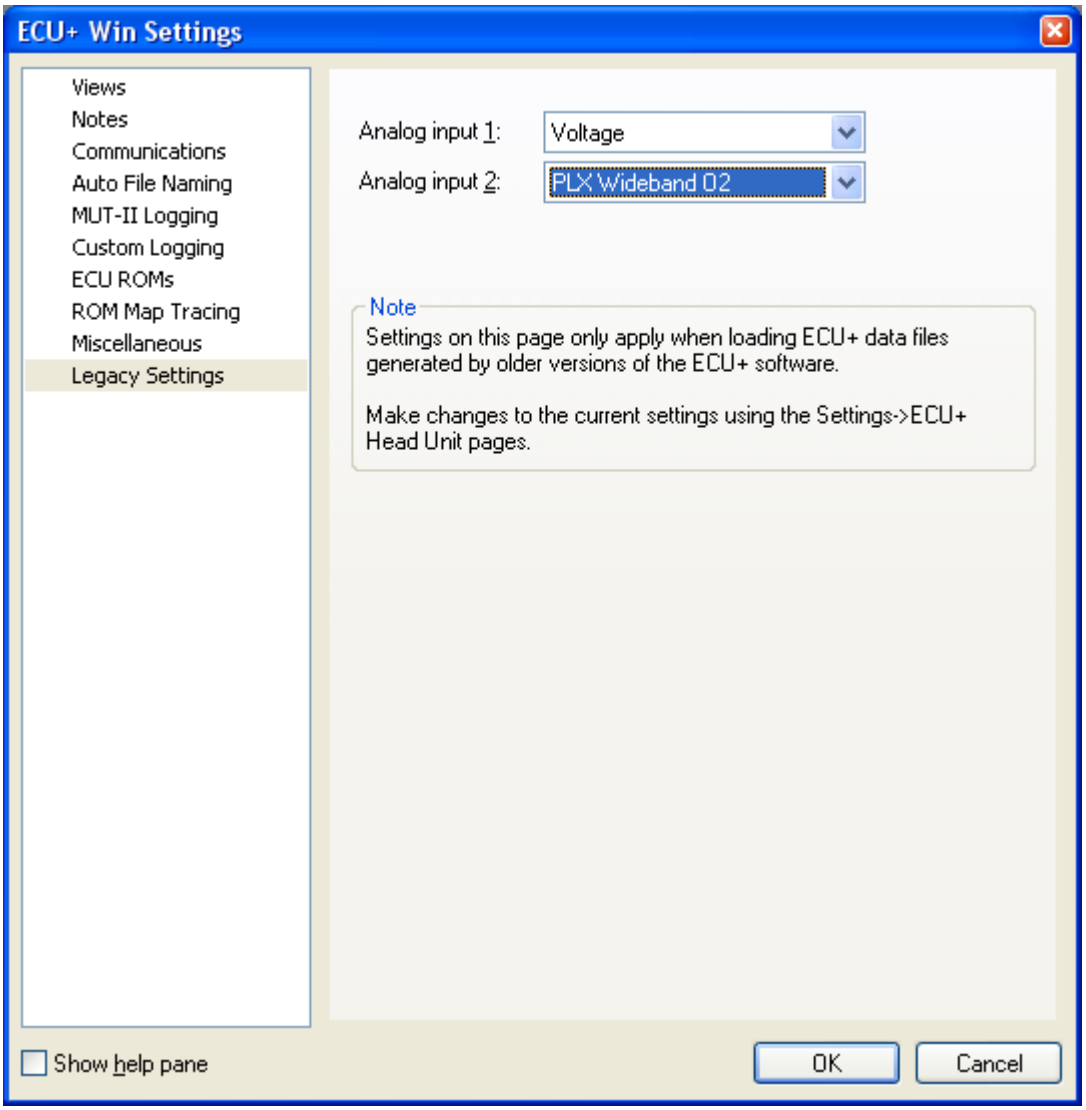


Illustration 43 - The legacy settings configuration dialog

The legacy settings configuration dialog configures how the ECU+ Win software should interpret the analog input 1 and 2 values when reading in capture files created with older versions of ECU+ Win. Previous versions of ECU+ Win don't "remember" how the analog input 1 and analog input 2 inputs were configured. Two dropdowns are available:

- *Analog input 1 – a*nalog input 1 can be used as a general-purpose 5 volt input, or it can be hooked to a EGT adapter which converts the signal from an EGT probe into a 5v signal. Select Voltage here if analog input 1 was used as a general-purpose input, or select 5v EGT Adapter if it was used with an adapter.
- *Analog input 2* – analog input 2 can be used as a general-purpose 5 volt input, or it can be hooked to a variety of supported wideband O2 sensors. Select Voltage here if analog input 2 was used as a general-purpose input, or select the wideband sensor kit if it was used with a wideband kit

7.7 The About Dialog

The About dialog, accessible from the menu item *Help->About ECU+ Win*, displays the version of the ECU+ Win software and a copyright notice.

7.8 The Views

When displaying the contents of a captured data file, the ECU+ Win software opens up multiple views of the data in the display section of the software. There are several types of views that the software can display. Some views display several values, some are graphs that can be scrolled around in, and some are just simple text displays. Let's take a look at them:

7.8.1 The Engine Monitor View

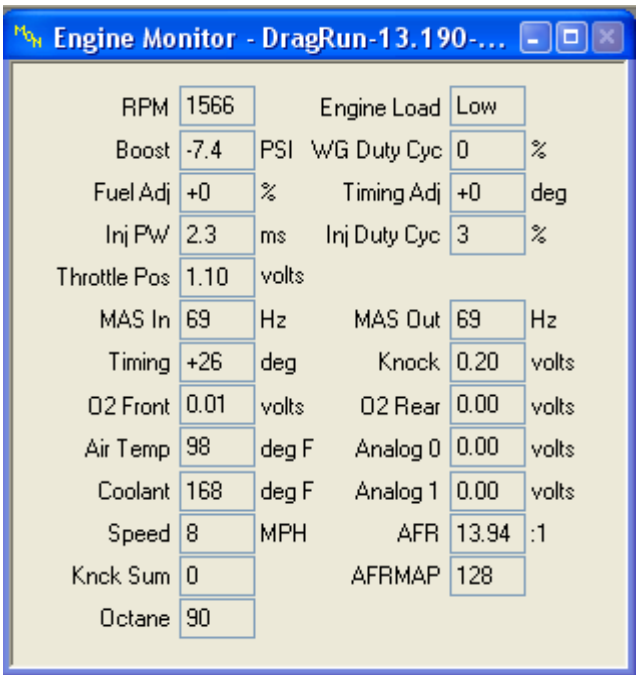


Illustration 44 - The engine monitor view

The engine monitor view (see Illustration 46) is your one-stop shop for everything that the ECU+ head unit knows about your engine's sensors. It displays a variety of information about the engine's sensors and what the head unit is doing. When doing a new capture of live data, the engine monitor displays the current data from the engine sensors in real time. When viewing a capture file from disk, the contents of the engine monitor reflect the cursor position on the current graph view. Thus, the engine monitor displays the current engine data at all times as a “snapshot” of what's going on.

Note: the engine monitor is the only view that can't be closed by clicking the 'X' in the upper right of the window.

7.8.2 The Notes View

Illustration 45 - The notes view

The notes view displays any notes associated with the current capture. You can edit a note's name or value by double-clicking on the field, making your change, and then pressing ENTER.

7.8.3 The Plot Color Legend View

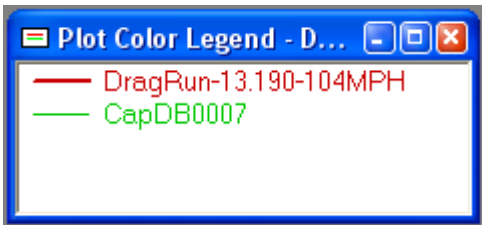
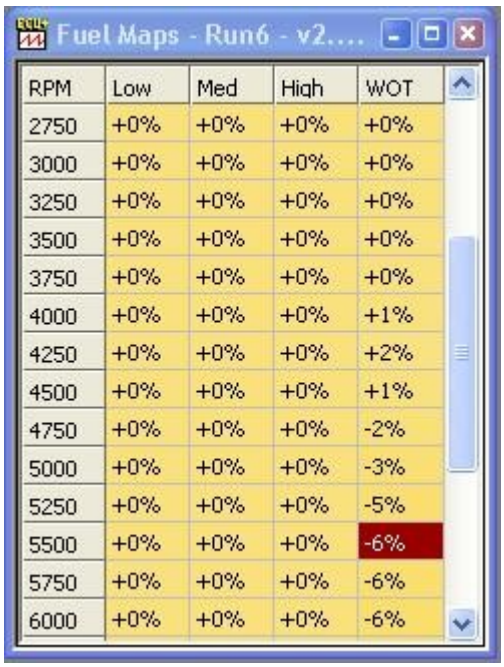


Illustration 46 - The plot color legend view

The plot color legend view (see Illustration 46) is a simple box that shows what colors are being used in the graph views when multiple files are overlaid.

7.8.4 The Fuel, Timing and Boost Maps Views



RPM	Low	Med	High	WOT
2750	+0%	+0%	+0%	+0%
3000	+0%	+0%	+0%	+0%
3250	+0%	+0%	+0%	+0%
3500	+0%	+0%	+0%	+0%
3750	+0%	+0%	+0%	+0%
4000	+0%	+0%	+0%	+1%
4250	+0%	+0%	+0%	+2%
4500	+0%	+0%	+0%	+1%
4750	+0%	+0%	+0%	-2%
5000	+0%	+0%	+0%	-3%
5250	+0%	+0%	+0%	-5%
5500	+0%	+0%	+0%	-6%
5750	+0%	+0%	+0%	-6%
6000	+0%	+0%	+0%	-6%

Illustration 47 - The fuel maps view

The fuel, timing and boost maps views show what ECU+ fuel, timing and boost maps were in place when the current capture was made.

If the capture is a live one, the current RPM and load values are used to highlight the fuel and timing map "cell" that's in-effect at the present time. If the capture has been loaded from disk, the highlighted cell corresponds to the cursor position. The view will auto-scroll so that the highlighted cell is always visible. Remember that the ECU+ head unit interpolates between fuel or timing map cell values, so the highlighted value doesn't necessarily reflect the exact fuel or timing offset that the ECU+ head unit is using – rather, it's the closest one. For example, if your engine RPM is 3125 RPM, the 3000 RPM map cell will be highlighted, when in fact the ECU+ head unit is actually using the value about half-way between 3000 and 3250 RPM.

If a second set of in use at a particular time, the views will change colors to reflect the map set that's selected.

7.8.5 The Graph Views

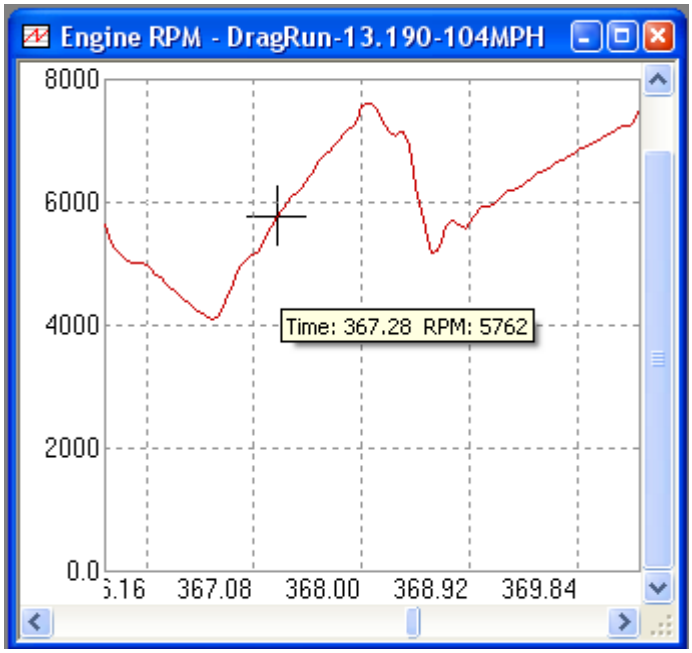


Illustration 48 - A graph view

The graph views are the most complex and powerful of the views to use. A graph view shows a single value from a capture in an X-Y plot, where the X axis is time, in seconds, and the Y axis is the single value. Graph views can be moved around, resized and zoomed. Let's see how these features work:

7.8.5.1 The Cursor and Moving Around

The cursor in a graph view (shown as a small cross that tracks the graph) represents the "current" spot on a graph. When you hover your mouse over a graph view, the cursor tracks the horizontal position of the mouse, and pops up a Windows tooltip showing the current X and Y axis values of the cursor on that graph. If there are any other graph views displayed, a similar tracking cursor is displayed which follows the mouse on all of the graphs at the same time. If you have a engine monitor view active, the values in the engine monitor reflect the current cursor position.

The ECU+ Win software includes a number of ways of scrolling and moving around within a view.

Table 5 shows the available keyboard and mouse actions. Some of these actions will be discussed later in more detail.

Action	Keyboard	Mouse
Cursor left	Shift+left arrow	Move
Cursor right	Shift+right arrow	Move
View start of capture	Home	
View end of capture	End	
Scroll left 1/20th of a page	Left arrow	X scrollbar arrows
Scroll Right 1/20th of a page	Right arrow	X scrollbar arrows
Scroll left 1 page	PgUp	X scrollbar
Scroll right 1 page	PgDn	X scrollbar
View larger Y values		Y scrollbar or scrollbar arrows
View lesser Y values		Y scrollbar or scrollbar arrows
X axis to next WOT region	Ctrl+T	
Zoom view's X axis in	F8	
Zoom view's X axis out	Shift+F8	
Zoom X axis to show entire view	Ctrl+F8	
Zoom view's Y axis in	F9	
Zoom view's Y axis out	Shift+F9	
Zoom Y axis to maximum size	Ctrl+F9	
View a subset of the current display		Left-click, hold, and draw a box
Shift last overlay left	Ctrl+left arrow	
Shift last overlay left ¼ of a page	Ctrl+shift+left arrow	
Shift last overlay right	Ctrl+right arrow	
Shift last overlay right ¼ of a page	Ctrl+shift+right arrow	
Cursor to next overlay	Tab	
Cursor to previous overlay	Shift+Tab	

Table 5 - Moving the cursor and scrolling

To move the cursor around on a graph view, simply move the mouse or use the keyboard arrow combinations Shift+Left or Shift+Right. If you'd like to see other parts of the graph, you can use the X scroll bar (or the keyboard right and left arrow keys) or the Y scroll bars. Note that the X axis of each graph view is synchronized with the X axis of all of the other graph views.

7.8.5.2 Zooming In And Out

Sometimes a graph view won't reflect exactly the X and Y ranges you'd like to see. The ECU+ Win

software gives you multiple ways of zooming in and out:

- The keyboard shortcuts F8, Shift+F8 and Ctrl+F8 are configured to zoom the X axis in, out, and to "full size" (as large as it can go), respectively.
- Similarly, F9, Shift+F9 and Ctrl+F9 zoom the Y axis.
- Other zoom options are available in the graph view context menu (more on this in the next section).
- Last but not least, you can use the left mouse button to manually zoom to a specific range – simply press and hold the left mouse button to draw a box around the region you want to see, and then release the button to zoom in.

7.8.5.3 Using the Graph Context (RMB) Menu

Each graph view has what's called a "Context" or right-mouse-button menu, which you can access by right-clicking on the graph. This invokes a new menu:

Illustration 49 - The graph view context menu

The context menu has several items. The first three items apply to the current graph only, while the remaining options apply to the ECU+ Win software. The menu items are as follows:

7.8.5.3.1 Zoom X Axis and Zoom Y Axis

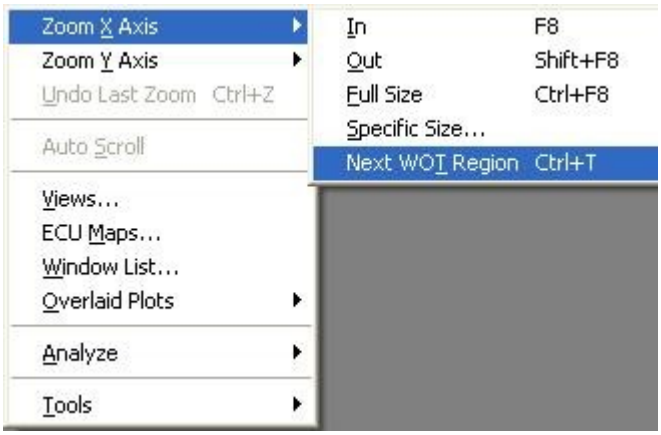


Illustration 50 - The zoom X axis sub-menu

Use these zoom X and Y axis menu items to zoom the X or Y axis of the graphs. Any X axis changes

apply to all graphs, while any Y axis changes apply to the current graph only. You can zoom each axis in, out or to full size, or you can set the X and Y axis values to a specific size. The X axis has an additional option, *Next WOT Region* which will zoom the X axis around the next WOT (wide open throttle) region past the cursor. The Y axis zoom also has an additional option, "Fit Visible," which will scale the Y axis to match the visible data in that graph.

7.8.5.3.2 Undo Last Zoom

This un-zooms the last zoom. The five most recent zooms are remembered, and can be un-done.

7.8.5.3.3 Auto Scroll

When capturing new data, this option is enabled. With the option checked, new data will cause all of the graph views to scroll rightward. If you un-check the option, the graph will not scroll with new data. Compare this option to the pause icon – pause throws away new data while keeping the graph stationary, while Auto Scroll keeps the graph stationary while continuing to acquire new data.

7.8.5.3.4 ECU Maps

This menu item is described in the section ROM Editing and Map Tracing on page 117.

7.8.5.3.5 Views and Window List

These menu items will be described in the section Setting Up Views on page 65.

7.8.5.3.6 Overlaid Plots

These menu items will be discussed in the section Overlaying Multiple Capture Files on page 66.

7.8.5.3.7 Analyze

These menu items will be described in the section Analysis Tools on page 69.

7.8.5.3.8 Tools

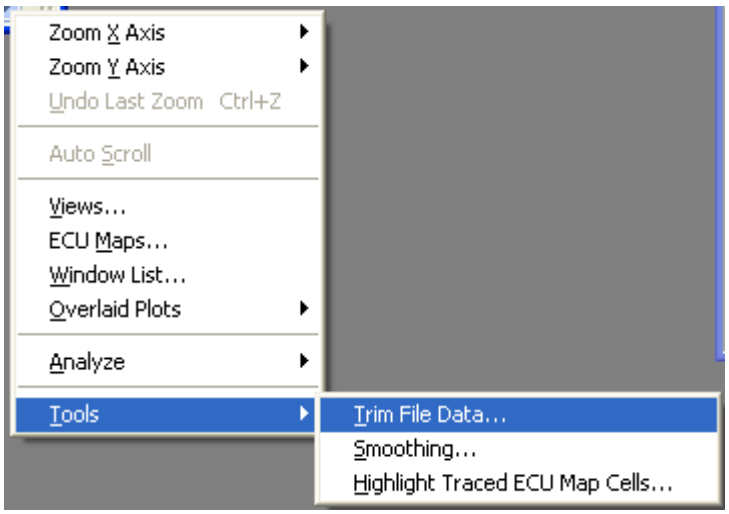


Illustration 51 - The tools menu

The tools sub-menu contains items for manipulating the displayed data.

Trim File Data. Use this when you have a long capture with only a short interesting section. To extract just the interesting region, use this option. A dialog will let you trim the data to just the currently visible region, or to any region that you specify. After trimming, you can save your trimmed data to a new .ecd file, if desired.

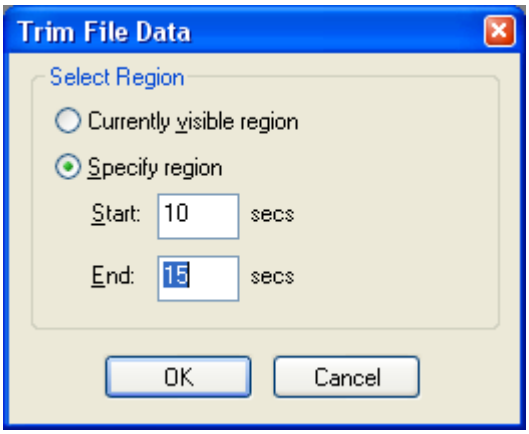


Illustration 52 - The trim file data dialog

Smoothing. This menu item allows you to change the amount of smoothing that is applied to a given graph value. Select the view type to be smoothed, and use the slider to set the smoothing. Smoothing values are remembered when you close ECU+ Win.

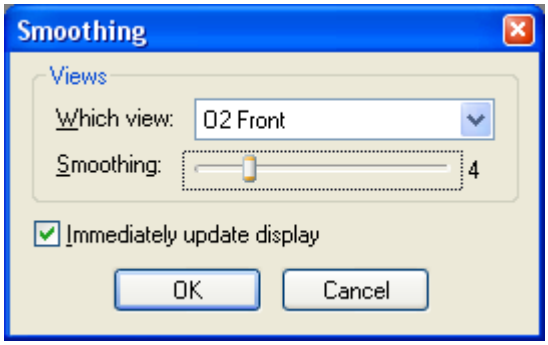


Illustration 53 - The smoothing dialog

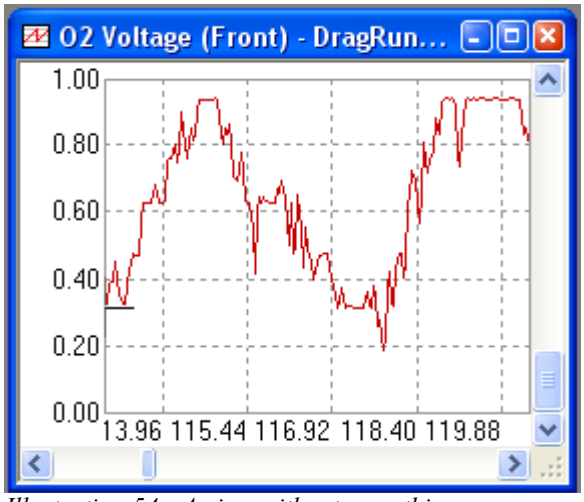


Illustration 54 - A view without smoothing

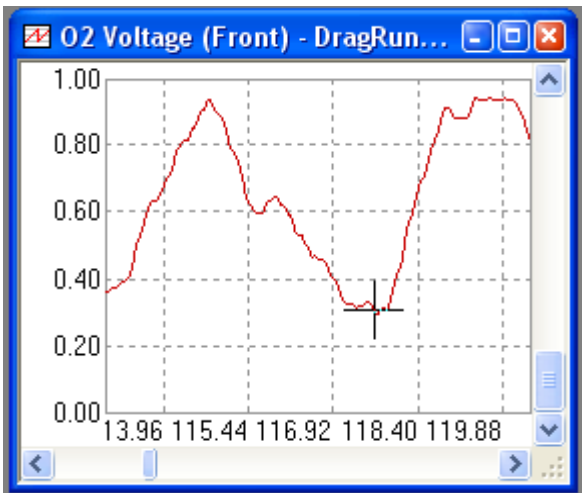


Illustration 55 - The view with a smoothing of 4

Highlight Traced Map Cells. This menu item allows is a toggle, and turns on or off stock ECU ROM map highlighting for a region of time. When turned on, you select a region of time, and the ECU+ Win software will highlight all of the stock ECU ROM map cells that were used during that region of time. See the below screenshot for an example.

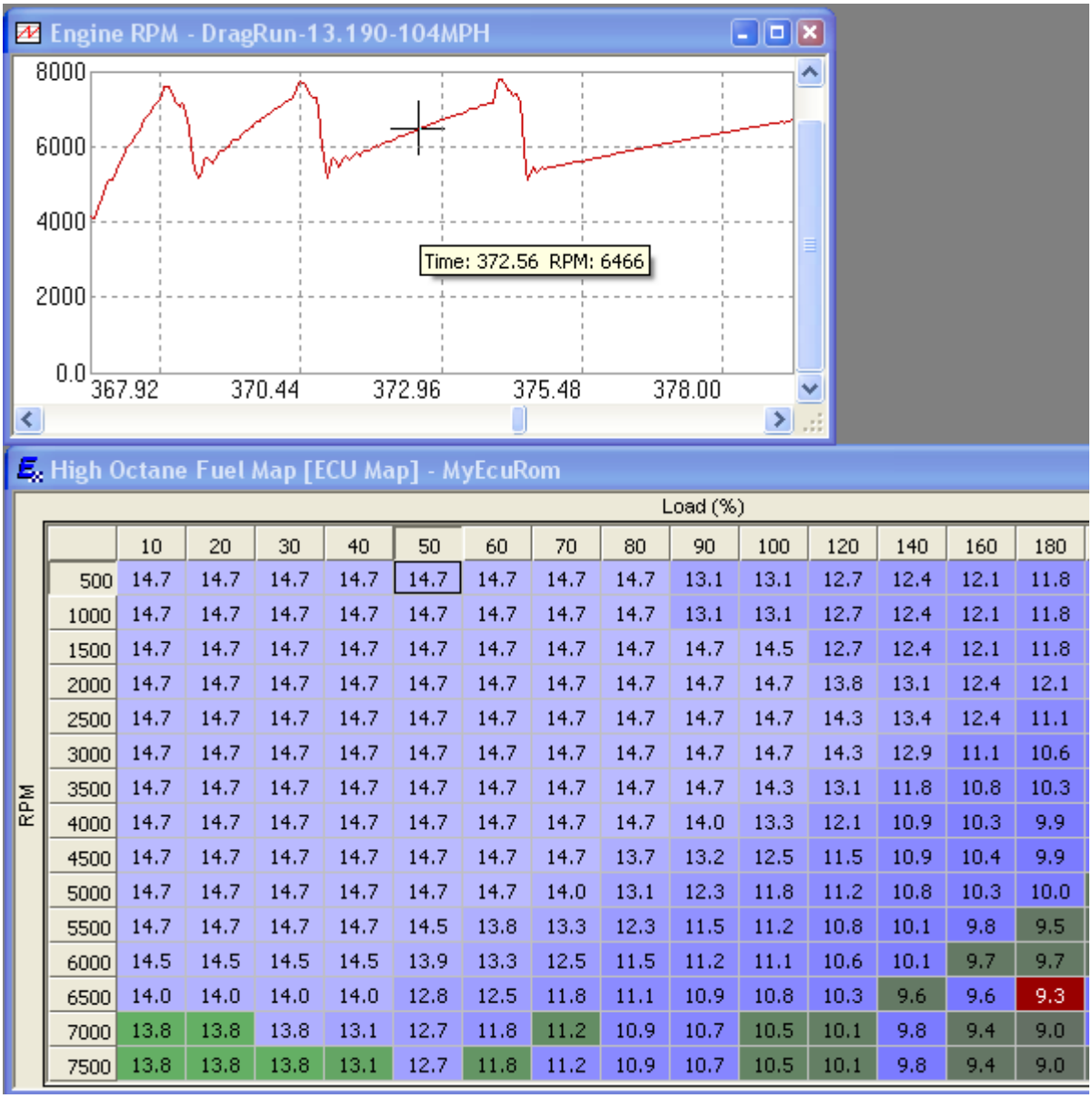


Illustration 56 - Stock ECU ROM map tracing in action

Note that this menu item is not present unless you've installed the version of ECU+ Win that allows ROM editing and map tracing.

7.8.6 The All In One Graph View

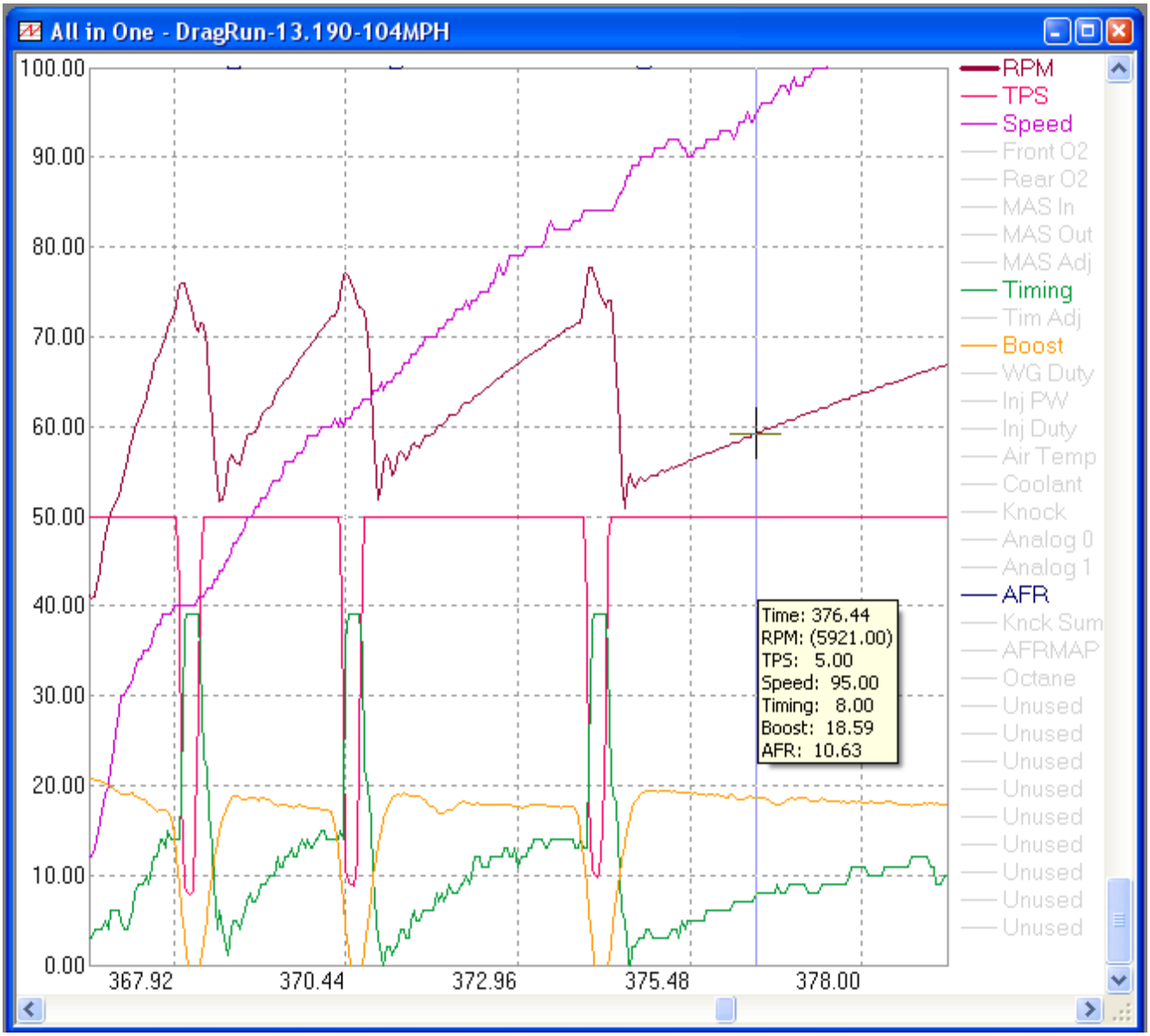


Illustration 57 - The all in one graph view

The all in one graph view is a special kind of graph view that displays multiple graphs from a single capture on one big, sometimes-confusing, sometimes-*very*-useful display. By displaying multiple signals on top of each other, it's possible to more easily see the relationship among engine sensor signals.

This view works much like the other graph views, with the familiar zooming and scrolling tools available like always on the right-click context menu. Only three things are different about this view:

1. Overlaying multiple captures isn't supported – the view is typically already cluttered enough.
2. The TAB key moves to the next graph rather than the next overlay.
3. There is a new *Graphs* menu item available on the right-click context menu.

The *Graphs* menu item brings up a new dialog box that allows you to configure the all in one graph view.

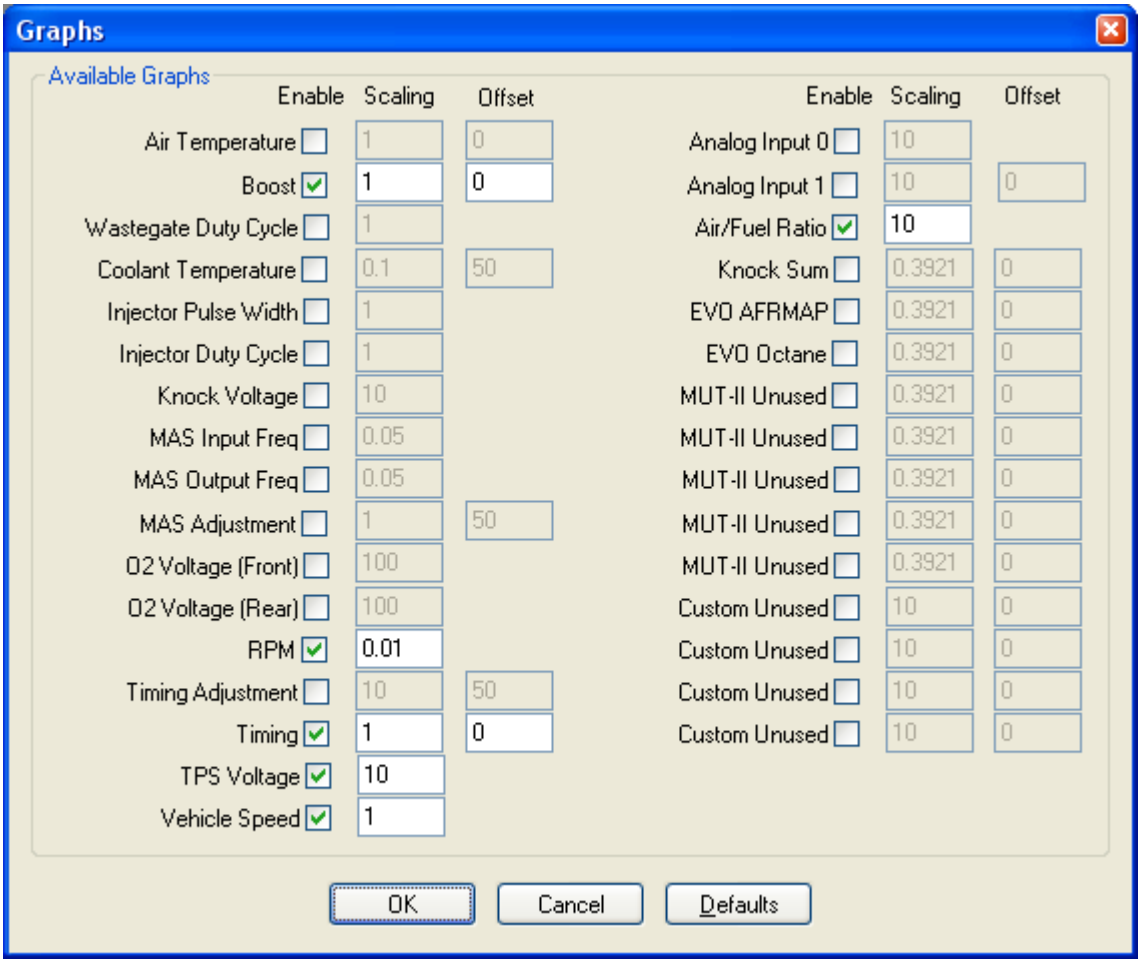


Illustration 58 - The graphs configuration dialog

This dialog shows all of the available signals that the all in one graph view can display. For each of the signals, a check box lets you turn on or off that graph. When a graph is displayed, you can configure the scaling and offset for that signal such that it displays correctly within the fixed-scale all in one graph view. The all in one graph view typically displays values from 0 to 100 on its Y axis. You'll want to set the scaling so that a particular signal makes sense when it's displayed in that range on the Y axis. For example, most analog signals are between 0 and 5 volts. For these, a scaling of 10 displays 5 volt signals as 50 – half way up the graph. The offset setting allows negative-going signals to be displayed. For example, the MAS adjustment normally ranges between +50 and -50 percent. Scaling this by one and offsetting by 50 puts zero adjustment in the center of the view, with deviations from zero going positive or negative away from the center. By tweaking the scaling and offset values for all of your signals, you can display a lot of information in the all in one graph view.

7.8.7 Setting Up Views

After starting a new capture or opening an existing capture file, the ECU+ Win software will open a collection of views to show you the data associated with the capture. Each view will open in a window in the display area of the software. You can then use the mouse or the Windows minimize and maximize buttons to resize each window. You best bet, though, is to display the views you're interested in, and then use the *Window->Auto Arrange* menu item to tile the windows.

If you've minimized a view window, or it's underneath another window and you can't see it, you can use either the Window menu, and then select a window by number to show it, or right-click on another window, and select *Window List*.

To close out a view window, click the 'X' in the window's upper right corner. To display a new view window, use the right-click context menu, and select *Views*. This will display a dialog that looks like this:

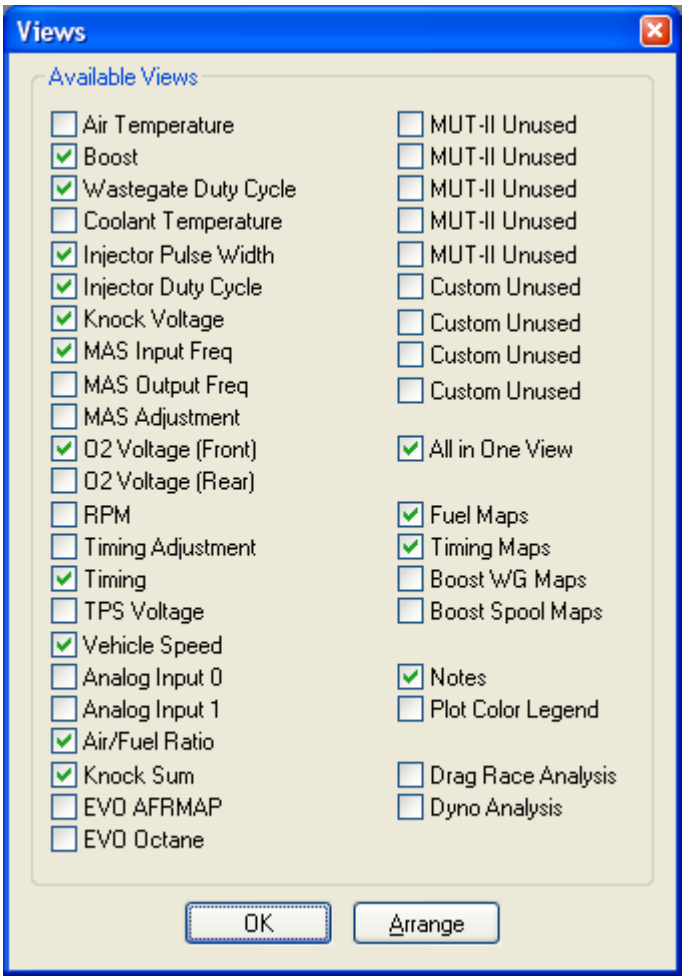


Illustration 59 - The views selector dialog

Click a checkbox on this dialog to activate or de-activate a view. If a view is visible, it has a check mark next to it. Minimized views have a grayed check mark. Once all of the interesting views are on-screen, use the *Arrange* button to auto-arrange all of the view windows.

The key points to remember are:

- Use *Context Menu->Views* to activate a view, or
- Use *Window->Window List* to un-minimize a view window or display it in front of the other windows

7.8.8 Overlaying Multiple Capture Files

When displaying a capture file, it's often handy to be able to view several capture files at the same time for comparison. The ECU+ Win software lets you do just that with the concept of "overlaid plots." Here's how it works:

1. Open up an existing capture file, and arrange the view windows the way you like them.
2. Use the right-click context menu on one of the graphs, then select *Overlaid Plots->Add New*.

The standard Windows file-open dialog will appear. Select another capture file.

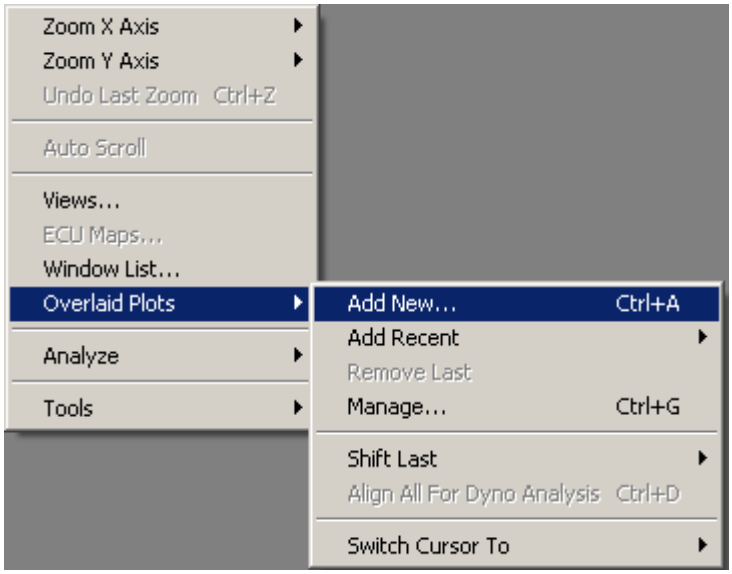


Illustration 60 - The overlaid plots menu

3. The ECU+ Win software will overlay this new capture directly on top of the original capture file, using a different color for each capture on the graph.
4. When you hover the mouse over the graph, the Windows tooltip will display the Y axis values for all of the overlaid captures. The current capture's value is in parenthesis. You can switch which capture is current with the TAB and shift-TAB keys.

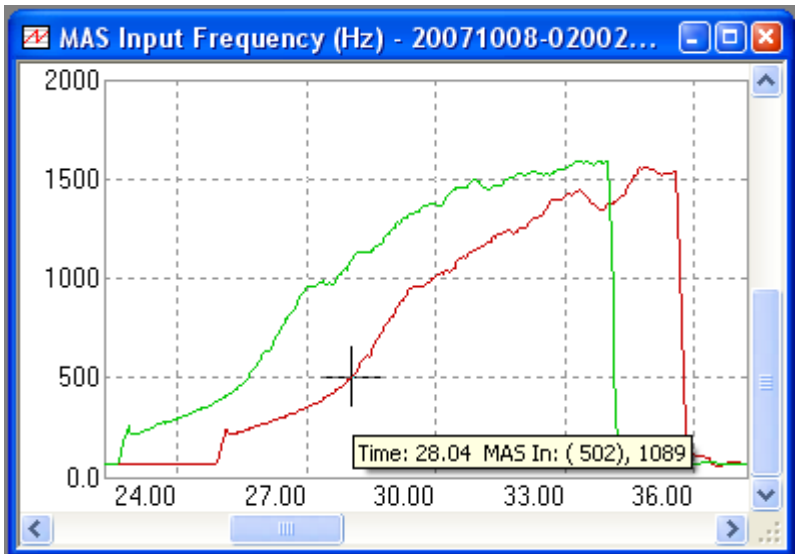


Illustration 61 - Two overlaid captures

5. Next use the *Shift Last* menu item (or the more-convenient *Align All for Dyno Analysis* menu item) to align the two plots in time:

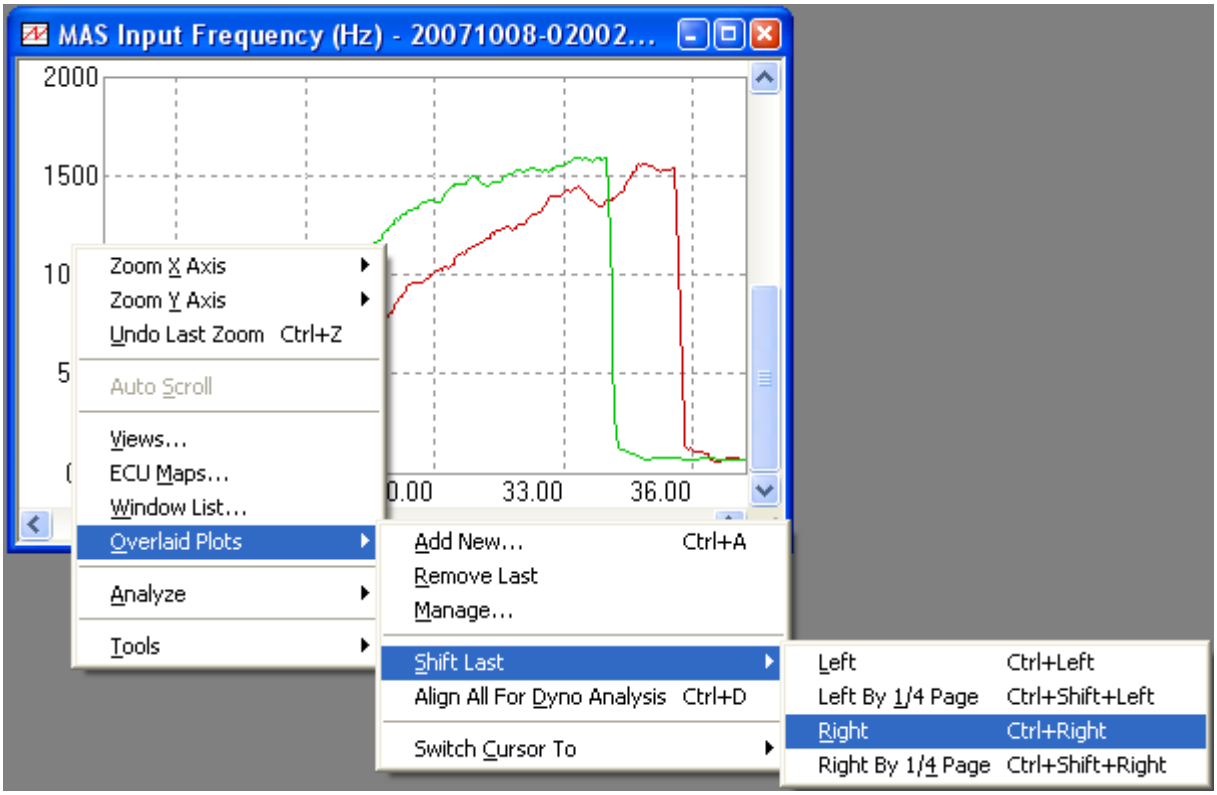


Illustration 62 - Aligning two captures

6. When you're all done, use the *Remove Last* menu item to remove the last capture from the graph views.
7. In addition to the menu items described above, the ECU+ Win software also includes a consolidated interface to the overlaid plots functionality. Select the *Manage* menu item to bring up this interface.

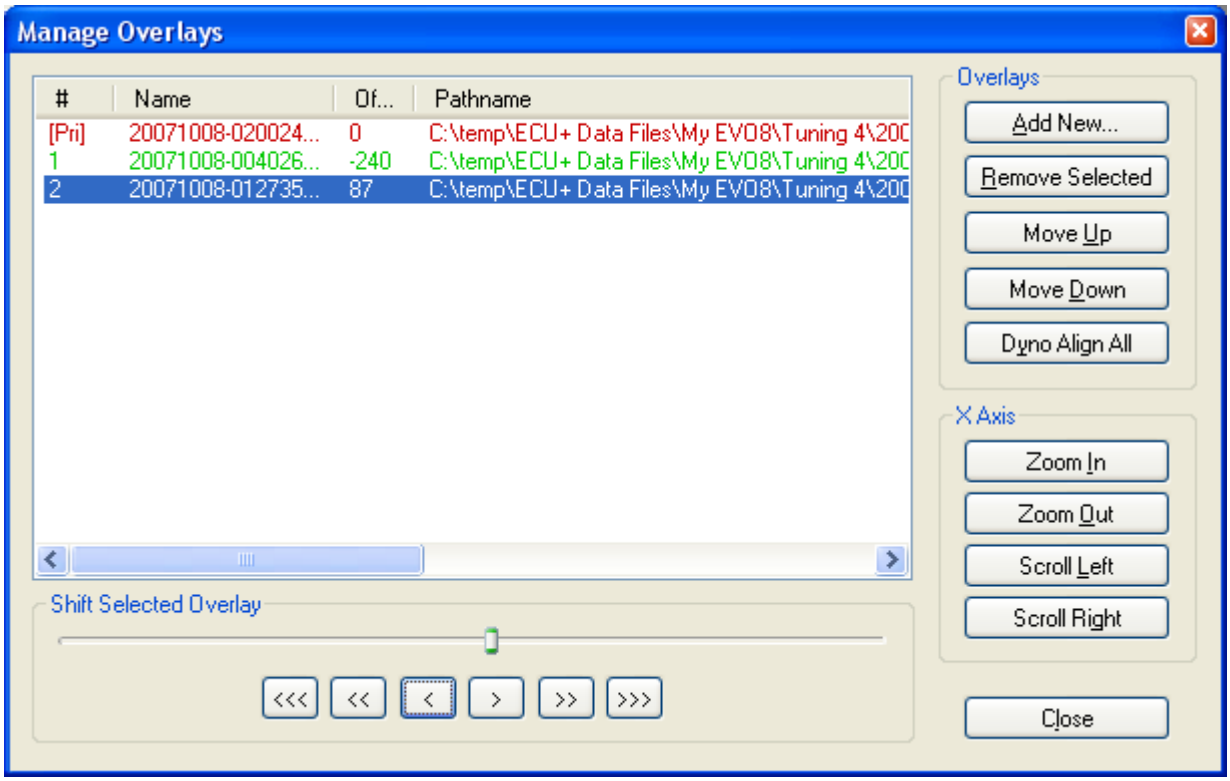


Illustration 63 - The manage overlays dialog

8. This interface lets you add, remove, reorder and align overlays very intuitively, as well as scroll around on the graphs. The slider is particularly handy - with a given overlay selected in the list, use the slider (or the arrow buttons) to shift overlays as desired. Updates on the graphs occur in real time.

Being able to overlay multiple captures is a very powerful technique to compare your vehicle's performance before and after a hardware modification or when tuning your car's fuel and timing via the ECU+ head unit. In the next section, we'll see some more powerful techniques with the Analysis Tools of the ECU+ Win software.

7.9 Analysis Tools

The ECU+ Win software offers two powerful analysis tools that will help you to tune your car for best performance. They are the Drag Race Analysis and the Dyno Analysis.

7.9.1 Drag Race Analysis

You can access the drag race analysis from the graph view context menu as *Analyze->As a Drag Race*. What this analysis tool does is to treat your capture as though it was a run down a ¼ mile dragstrip, and calculates the elapsed time, vehicle MPH and distance 25 times a second. Additionally, it'll display typical drag race statistics, like 60' time and ¼ mile trap speed.

To use this tool, select *Analyze->As a Drag Race* from the graph view context menu. You'll see a dialog like Illustration 66:

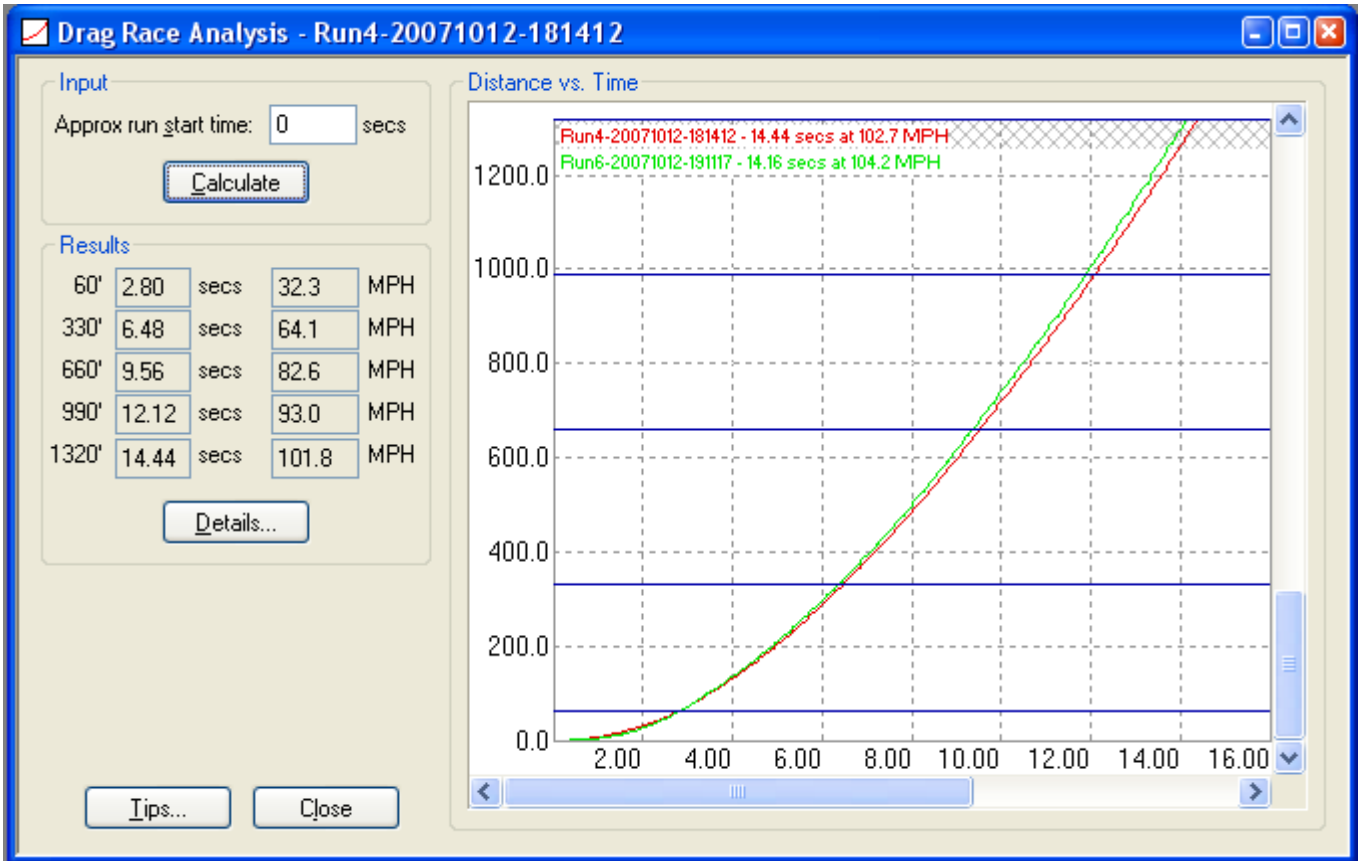
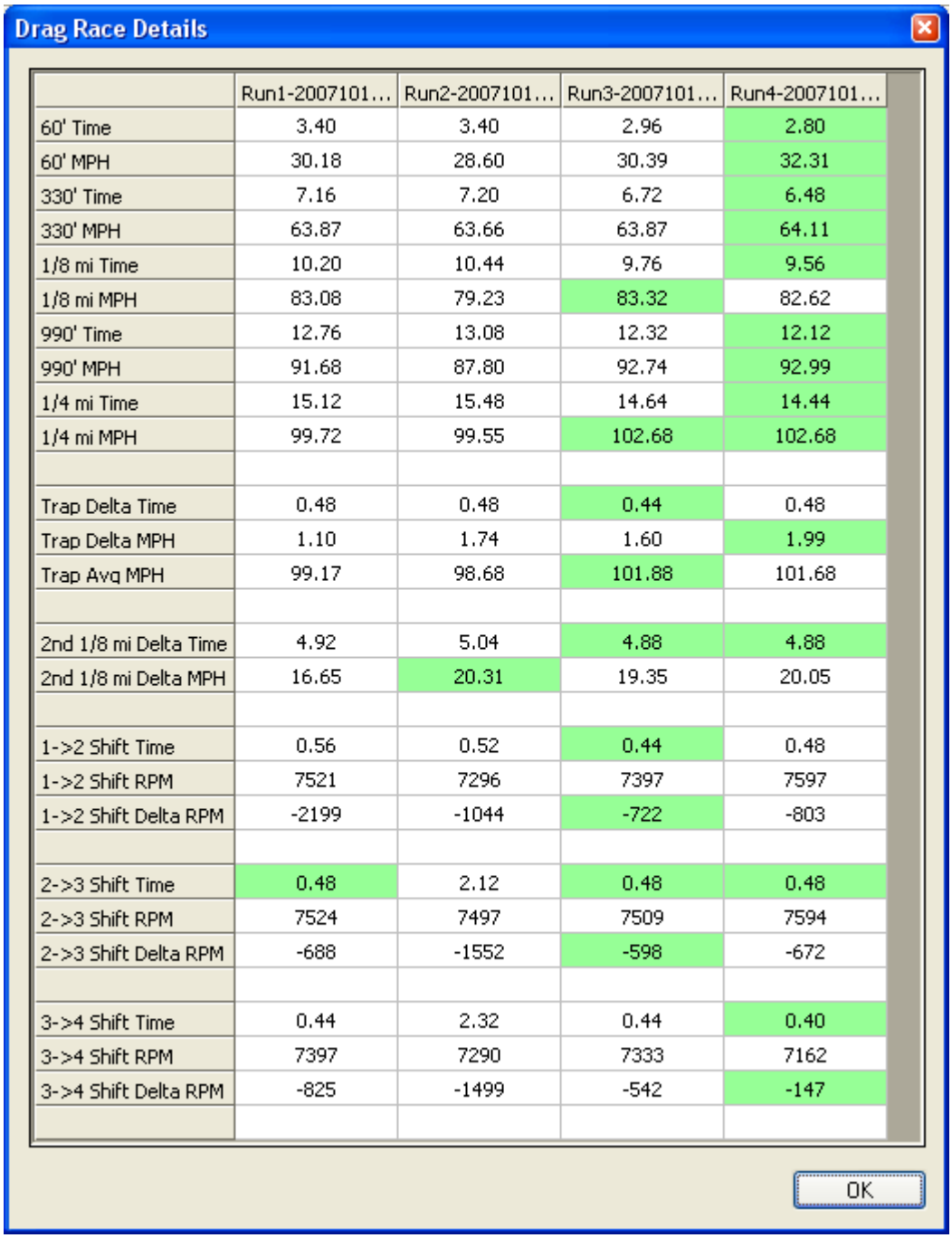


Illustration 64 - The drag race analysis view

With the dialog visible, enter in the approximate start time of the drag race and click the *Calculate* button. The ECU+ Win software will try to find a spot past the start time in which the car was over 60 MPH (so the starting time doesn't have to be very accurate at all), and then search backwards to find the time when the car was stopped. It'll treat the stopped time as the start of the drag race and compute the vehicle's ¼ mile performance. If multiple captures are overlaid, an analysis will be done on each and the distance vs time plot will show all captures. To toggle the Results pane between multiple overlaid captures, right-click on the graph and select *Switch Cursor To->Next Plot*.

In addition to the distance vs time graph, the ECU+ Win software also includes two other features on the drag analysis window:

- The *Tips* button, which pops up a collection of help with drag racing, including tips and tricks for getting the best times.
- The *Details* button, which displays additional ¼ mile statistics. This is especially handy with multiple overlaid plots, as this shows the statistics for all plots side-by-side, and highlights the best run in multiple categories. A screenshot is shown below.



	Run1-2007101...	Run2-2007101...	Run3-2007101...	Run4-2007101...
60' Time	3.40	3.40	2.96	2.80
60' MPH	30.18	28.60	30.39	32.31
330' Time	7.16	7.20	6.72	6.48
330' MPH	63.87	63.66	63.87	64.11
1/8 mi Time	10.20	10.44	9.76	9.56
1/8 mi MPH	83.08	79.23	83.32	82.62
990' Time	12.76	13.08	12.32	12.12
990' MPH	91.68	87.80	92.74	92.99
1/4 mi Time	15.12	15.48	14.64	14.44
1/4 mi MPH	99.72	99.55	102.68	102.68
Trap Delta Time	0.48	0.48	0.44	0.48
Trap Delta MPH	1.10	1.74	1.60	1.99
Trap Avg MPH	99.17	98.68	101.88	101.68
2nd 1/8 mi Delta Time	4.92	5.04	4.88	4.88
2nd 1/8 mi Delta MPH	16.65	20.31	19.35	20.05
1->2 Shift Time	0.56	0.52	0.44	0.48
1->2 Shift RPM	7521	7296	7397	7597
1->2 Shift Delta RPM	-2199	-1044	-722	-803
2->3 Shift Time	0.48	2.12	0.48	0.48
2->3 Shift RPM	7524	7497	7509	7594
2->3 Shift Delta RPM	-688	-1552	-598	-672
3->4 Shift Time	0.44	2.32	0.44	0.40
3->4 Shift RPM	7397	7290	7333	7162
3->4 Shift Delta RPM	-825	-1499	-542	-147

Illustration 65 - The drag race details screen

You can use the drag race analysis on a real drag strip (or a deserted road) to see how your car was doing at various points in the run, and to compare the car's performance before and after a modification.

Some considerations to keep in mind when using the drag analysis:

- The calculations use the vehicle's speed sensor, so if you've got non-stock gearing or tires, the analysis may not match exactly what you'd get from a real drag strip run.
- The calculations assume no tire spin, which is fairly accurate for AWD cars. FWD cars will be less accurate.
- The clutch switch input is used, if applicable to measure shift times. If the clutch switch input isn't used, dips in TPS are used to measure shift times. The clutch switch input is much more

accurate.

7.9.2 Dyno Analysis

The dyno analysis tool, accessible from the graph context menu (*Analyze->As a Dyno Pull*) or from the dyno icon on the toolbar, is perhaps the most useful analysis tool in the ECU+ Win software. What the dyno analysis tool does is to treat a short, single-gear acceleration as a dyno pass. From the data in the capture file, ECU+ Win will calculate your engine's horsepower and torque and plot it in an X-Y graph that can be zoomed or printed just like the other views. Here's how to use it:

1. Find a deserted back road where you can do some test runs. The road should be flat and straight. Use a road sign or marker on the road as a "launch" spot.
2. Start a new capture on the ECU+ Win software.
3. Drive your car along the back road in 2nd (or 3rd, if you have access to a drag strip) gear, and "lug" the car at the lowest possible RPM. As soon as you pass the launch spot, floor the accelerator and let the car accelerate to redline. Then coast back down and pull off of the road.
4. Stop the capture and save the file. It may be useful to name the file so that you can easily recognize it later ("20090301 Run3 – Boost at 19 PSI.ecd").
5. Later at home, or after stopping the vehicle, open this new capture in the ECU+ Win software, and manually zoom the X axis so that just the acceleration time is visible. This will be easy to find, as the TPS sensor will show the car at WOT, and pretty much everything will be increasing with time. The X axis *Zoom to Next WOT* menu item will most likely zoom to exactly the region you're interested in.

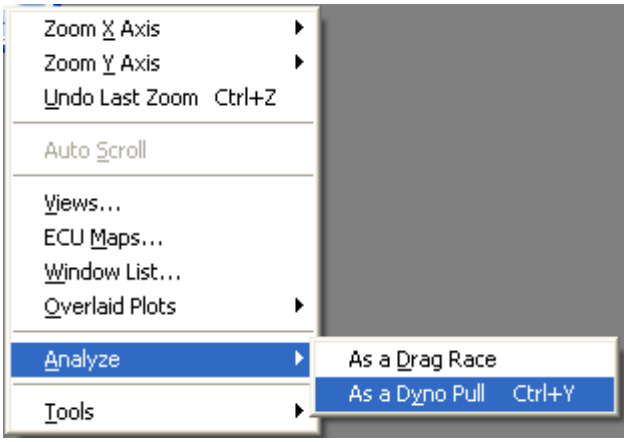


Illustration 66 - The dyno analysis menu item

6. Open the dyno analysis as shown in Illustration 66. The ECU+ Win software will have filled in the start and stop times from your zoomed graph. Update the other inputs on the left side, and press the *Calculate* button. The ECU+ Win software will now display your car's horsepower and torque graph, just as though you had done a run on a real dyno. See Illustration 67 for a sample graph.

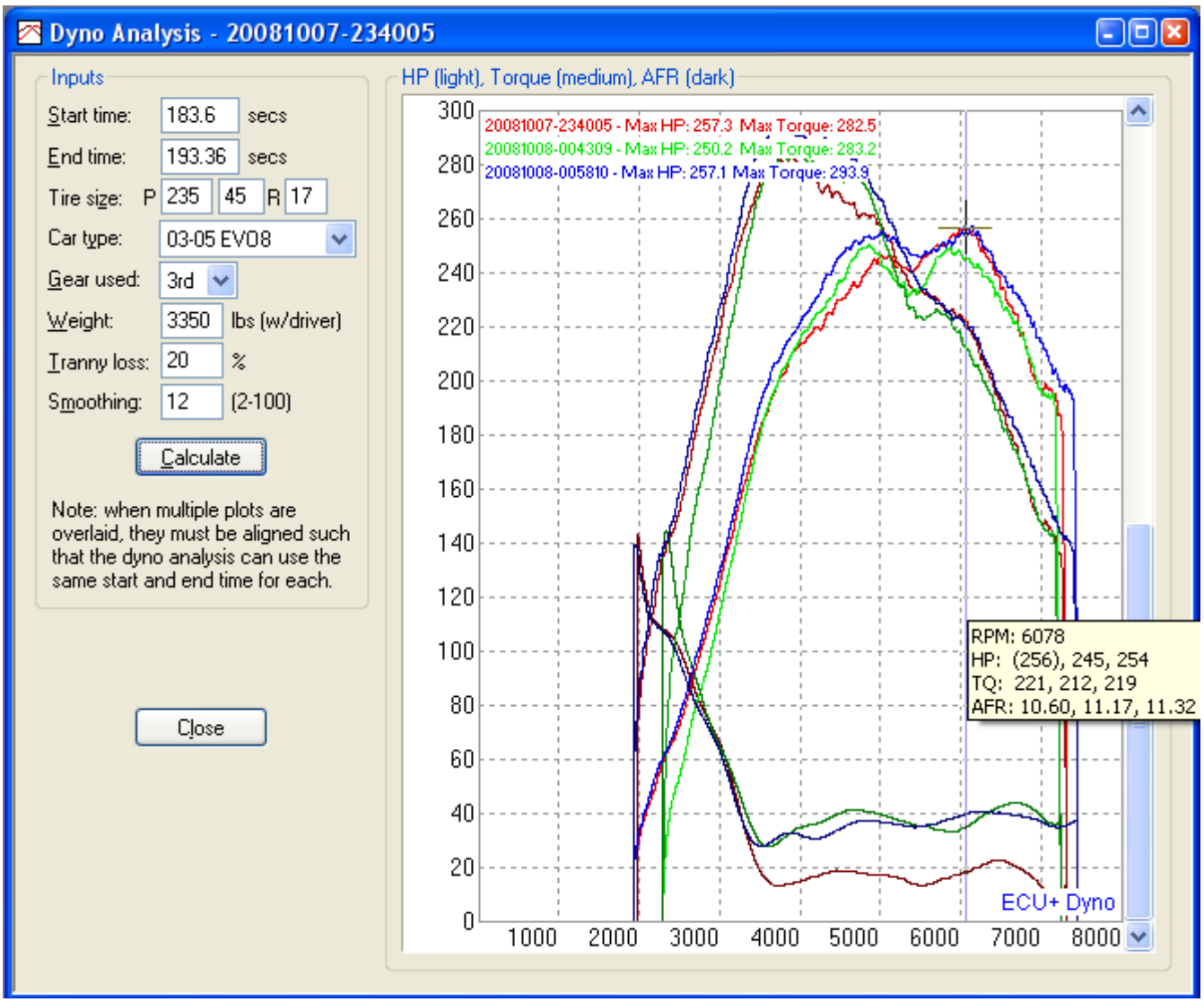


Illustration 67 - A dyno analysis

With the dyno analysis done, you can zoom the dyno graph with the mouse, or right-click for a context menu that has the usual zoom capabilities. You can also print the dyno analysis with the *File->Print* menu item. Use the *Close* button to close the dyno analysis.

The dyno analysis uses the engine's RPM to compute horsepower and torque. Because the vehicle's speed (and thus acceleration, and thus torque and horsepower) is calculated based on the car type, car weight, gear used, and tire size, it's important to set these parameters consistently from one dyno run to the next in order to see reliable plots.

Some things to consider when doing a dyno analysis:

- The dyno analysis uses the *Tranny loss* parameter to compute the engine's crank horsepower from the measured (wheel) horsepower. If you'd like to plot wheel horsepower, put zero in this field.
- The *Smoothing* parameter smooths the engine RPM before doing the dyno calculation. Larger numbers give smoother plots. Generally, values in the 8-20 range work best.
- When the mouse is hovered over a point on the dyno graph, a tooltip is shown with the current HP, torque and air/fuel ratio. Additionally, the cursor “tracks back” to the main ECU+ Win

views such that the cursor on, say, the RPM view corresponds to the time that your car was making the HP indicated on the dyno analysis.

- The software plots HP, torque and air/fuel ratio on the graph. 10:1 AFR is shown as 20 on the graph, and 11:1 is shown as 40. This was done to increase resolution. If you hover over a given point in the graph, the correct air/fuel ratio is shown.
- The ECU+ Win software will remember your dyno analysis parameters from one session to the next, so it makes sense to put in the correct values for tire size and weight.
- When testing the effects of a performance modification (like a new air filter, or a change in the fuel maps), it's important to do your "dyno runs" in similar weather, and always use the same launch point on the road.
- You can overlay two or more captures, and then do a dyno analysis as shown in Illustration 67. The dyno analysis will show you the dyno results from each of the captures, all overlaid on the same dyno plot. This is incredibly useful when tuning your car. The only caveat is that you must align the two overlaid graphs such that the dyno analysis time is the same for each. As an example, if graph 1 did a second gear acceleration starting at 32 seconds, and graph 2 started at 28 seconds, you'll need to move the second overlay to the right by 4 seconds. Use the graph context menu item *Overlaid Plots->Shift Last->Align For Dyno Analysis* to do this automatically.
- The dyno numbers are mathematically correct, but may not exactly reflect the numbers that you'll get on a true vehicle dyno. Nonetheless, what's important is not the absolute horsepower and torque numbers, but instead that you can see whether changes you make to your vehicle increase or decrease its performance.

7.10 Configuring the ECU+ Head Unit

With the exception of the main datalogging display, you'll be using the head unit configuration screen the most of any part of the ECU+ Win software. The head unit configuration screens allow you to configure the parameters stored in the silver box, and those parameters are used to tune your car and to set up the special features of the ECU+.

To start the ECU+ Win configuration dialog, either click the hammer icon in the toolbar, or use the *Settings->ECU+ Head Unit* menu item.

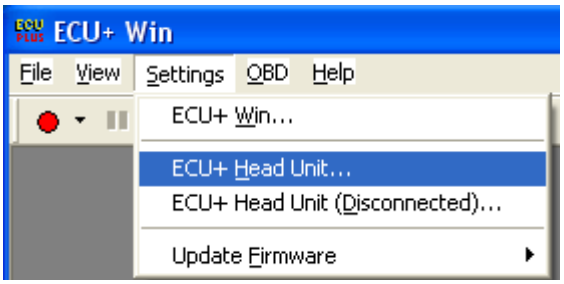


Illustration 68 - The head unit configuration menu items

A typical head unit settings dialog is shown below.

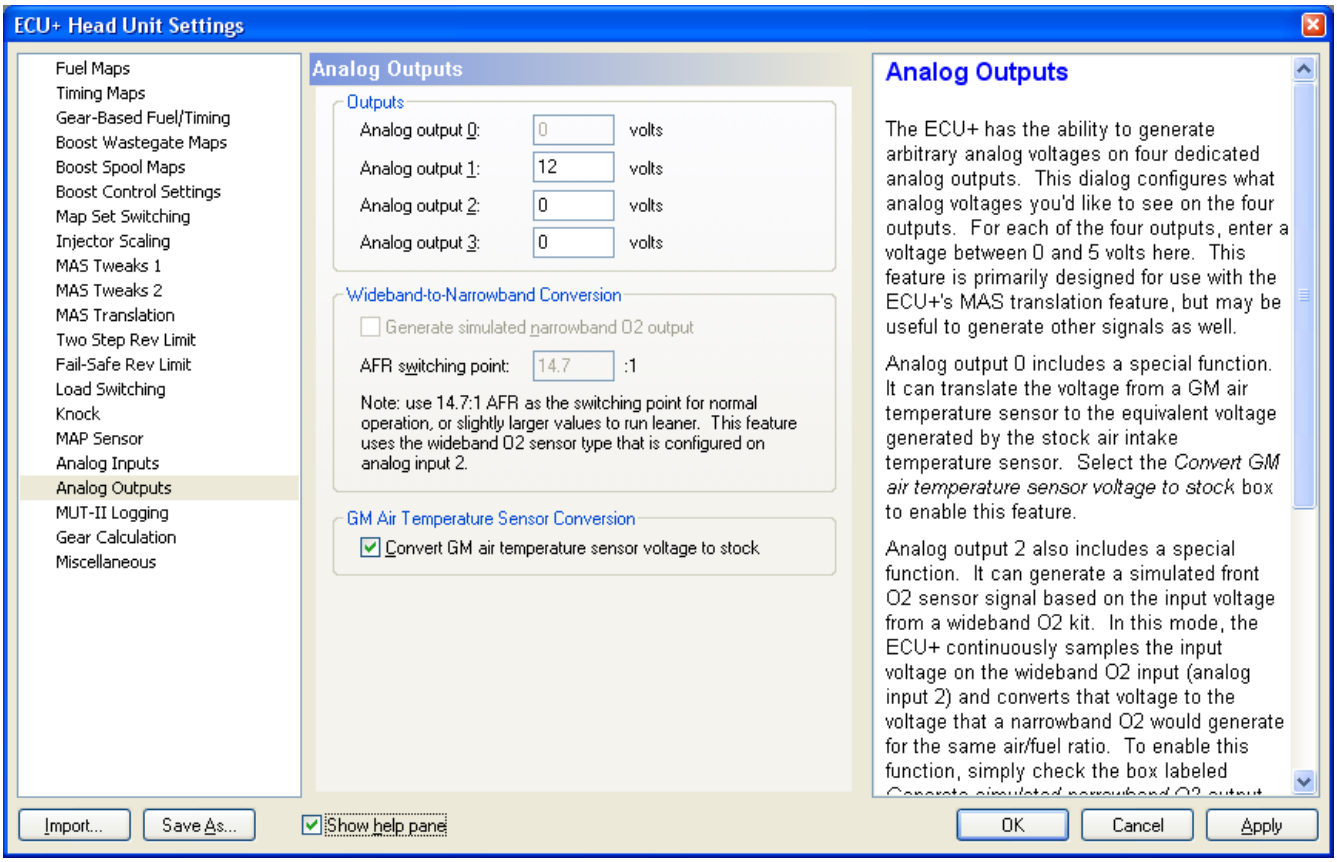


Illustration 69 - A typical head unit settings dialog

The left pane of the dialog shows a category of settings – click a new category to bring those settings into view. You'll make changes by editing values in the center pane of the dialog. The rightmost pane contains online help for each configuration option; it can be toggled on and off with the *Show help pane* button.

After you make changes to a setting, either press the *Apply* button to store all changed settings into the ECU+ silver box, or *Ok* to store the settings and close the dialog box.

Let's take a look at the configuration screens in turn:

7.10.1 The Fuel Maps

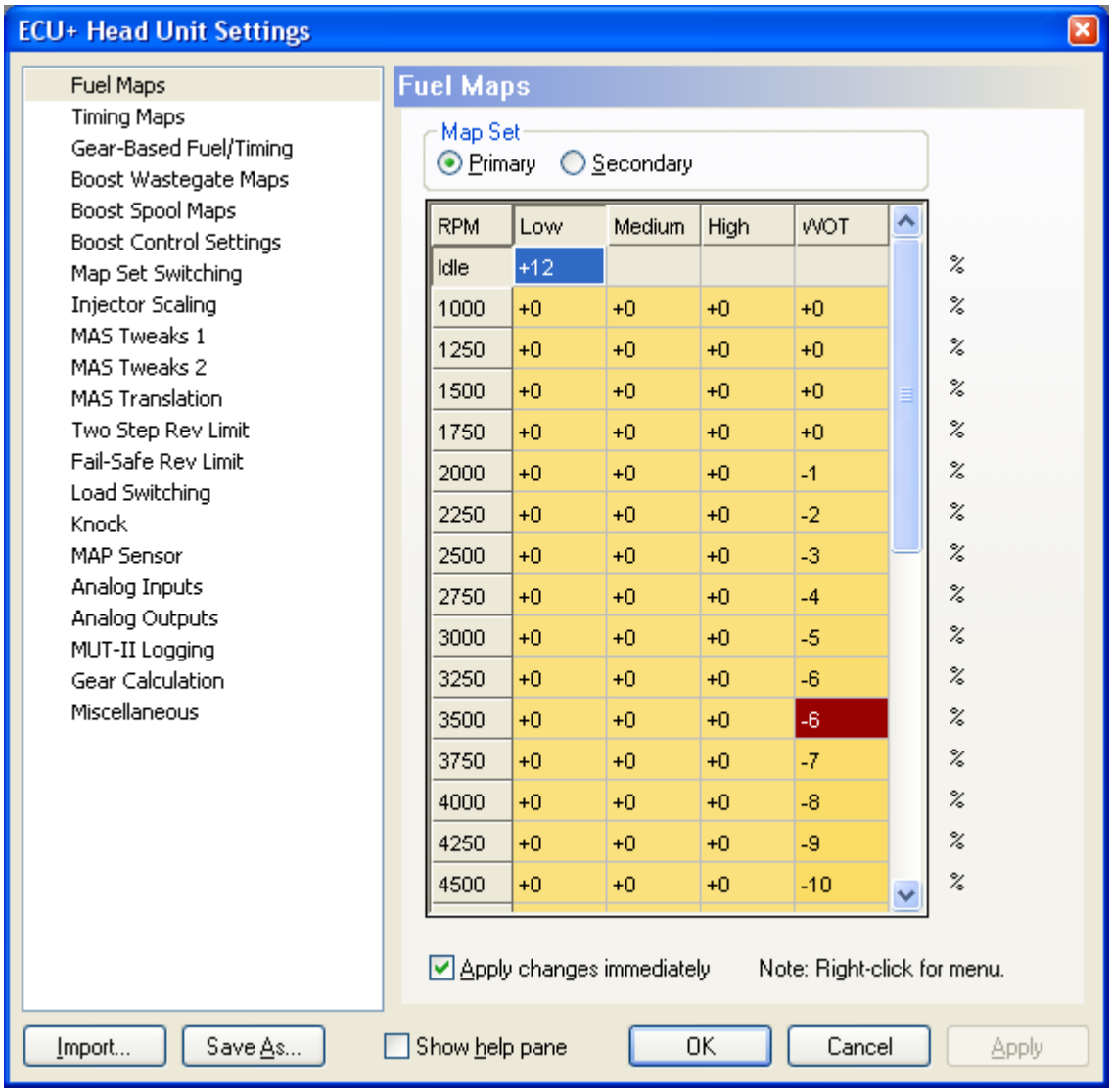


Illustration 70 - The fuel maps dialog

The fuel maps dialog lets you configure the ECU+ head unit's fuel maps. The fuel maps tell the ECU+ head unit how to modify the engine's mass air sensor (MAS) signal. To increase or decrease the fuel flow to your engine, the ECU+ head unit modifies this MAS signal. The MAS (in the engine bay) produces a frequency proportional to the amount of air entering the engine. By modifying the MAS signal, the ECU+ head unit "lies" to the stock ECU about how much air is entering the engine, which causes the stock ECU to change its fuel flow proportionally – less air means less fuel.

The fuel maps contain 133 different "cells" - one for idle and 33 cells for low, medium, and high engine "loads," as well as 33 cells for wide-open throttle conditions. Each cell represents how the ECU+ head unit should modify the MAS frequency when the engine is running at that particular load and RPM combination. RPMs range from 1000 RPM up to 9000 RPM.

An example: let's say your car is accelerating at wide open throttle, and the engine is at 4000 RPM, and the cell value for 4000 RPM/WOT is -10%. If the engine's MAS frequency is 2200 Hz (representing some fixed amount of air flow), the ECU+ head unit will change this frequency to 90% of 2200 Hz, or 1980 Hz, and pass this new frequency on to the stock ECU. The stock ECU will then reduce the corresponding fuel flow by 10%, thus causing your car to run approximately 10% leaner.

To set a given fuel map cell to a particular value, select a cell by clicking it with the mouse, type in a number and press ENTER. You can also apply operations to multiple cells. First, select multiple cells as shown in the table below.

Action	Mouse	Keyboard
Select a row of cells	Click an RPM header	
Select a column of cells	Click a load header	
Select all cells	Click the word *RPM*	
Select a rectangular block of cells	Click the first cell, then shift+click the last cell	With the cursor in the first cell, use shift+up/down/left/right arrows
Select multiple disconnected cells	Click the first cell, then ctrl+click another cell	

Table 6- Available shortcuts to select multiple cells

With multiple cells selected, use the right-click context menu to alter all of the selected cells.

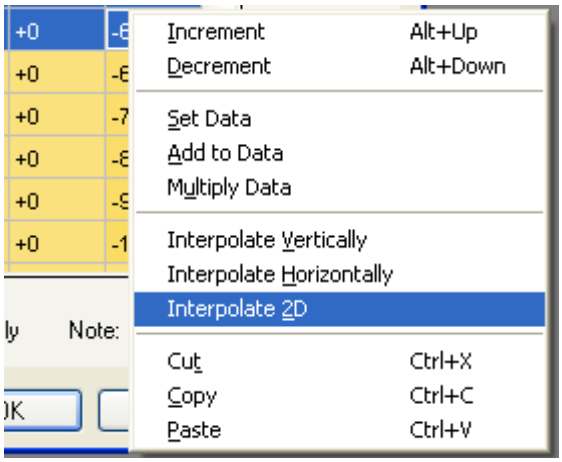


Illustration 71 - The fuel maps context menu

The options on this menu are:

- *Increment* – increases all of the selected cells by 1.
- *Decrement* – decreases all of the selected cells by 1.
- *Set Data* – sets all of the selected cells to a value you enter.
- *Add to Data* – adds a value you enter to all of the selected cells.
- *Multiply Data* – multiplies all of the selected cells by a value you enter.
- *Interpolate Vertically* – interpolates the intermediate values of a column of cells based on the first cell and last cell values. As an example of how this works, suppose a column of 5 contiguous cells are selected, and that the cells contain the values 0, 12, 15, 17 and 8 (top to bottom). Interpolating this group vertically will fill in the values between 0 and 8, leaving the cells with 0, 2, 4, 6 and 8. This feature is useful when creating a increasing or decreasing range of cells. Note that if multiple columns of cells are selected, this option will vertically interpolate each column independent of the others.

- *Interpolate Horizontally* – similar to *Interpolate Vertically*, but works on rows of cells.
- *Interpolate 2D* – this is similar to the other interpolate options, but this interpolates a square group of selected cells in two dimensions, using the "corner" values as the limits. An example illustrates this.

RPM	Low	Medium	High	WOT
Idle	+0			
1000	+0	+0	+0	+4
1250	+0	+0	+0	+0
1500	+0	+0	+0	+0
1750	+0	+0	+0	+0
2000	+0	+0	+0	+0
2250	+10	+0	+0	+14
2500	+0	+0	+0	+0
2750	+0	+0	+0	+0
3000	+0	+0	+0	+0
3250	+0	+0	+0	+0
3500	+0	+0	+0	+0
3750	+0	+0	+0	+0

Illustration 72 - Fuel maps before 2D interpolation

RPM	Low	Medium	High	WOT
Idle	+0			
1000	+0	+1	+3	+4
1250	+2	+3	+5	+6
1500	+4	+5	+7	+8
1750	+6	+7	+9	+10
2000	+8	+9	+11	+12
2250	+10	+11	+13	+14
2500	+0	+0	+0	+0
2750	+0	+0	+0	+0
3000	+0	+0	+0	+0
3250	+0	+0	+0	+0
3500	+0	+0	+0	+0
3750	+0	+0	+0	+0

Illustration 73 - Fuel maps after 2D interpolation

- *Cut*, *Copy* and *Paste* – the standard Windows clipboard operations. These can operate on a rectangular region of cells as well as a column or row of cells, and allow copy and pasting cell values within a table or between tables.

Each fuel map cell can take on values from -50% to +50%, where negative numbers decrease your engine's fuel flow (causing the engine to run leaner), and positive numbers increase it (causing the engine to run richer).

Note that the ECU+ head unit interpolates values between the discrete levels in the fuel map table. That is, if your fuel map cell value is -10% at 6000 RPM and -12% at 6250 RPM, and your engine is running at 6125 RPM, the ECU+ head unit will adjust the fuel by -11%.

The ECU+ head unit includes two independent sets of fuel maps, a "Primary" one and a "Secondary" one. The primary set of fuel maps is used by default inside the head unit, while the secondary set is switched to based upon a user-configurable set of parameters. To edit the primary fuel maps, make sure that the *Primary* radio button is selected. To edit the secondary fuel maps, select the *Secondary* radio button.

Also on the fuel maps dialog is a check box that tells the ECU+ Win software to send fuel map changes to the ECU+ head unit immediately. If this isn't checked, values are only sent over when you click *Ok* or *Apply*. Changes that haven't been applied are displayed in a bold font.

7.10.2 The Timing Maps

The timing maps dialog lets you configure the ECU+ head unit's timing maps. The timing maps tell the ECU+ head unit how to modify the engine's cam and crank angle sensor signals. When the ECU+ head unit modifies the cam and crank angle sensor signals, it causes the stock ECU to fire the spark plugs either sooner or later than stock, thus affecting the engine timing.

The timing maps contain the same 133 cells that the fuel maps do, corresponding to a given engine load

and RPM, and are configured exactly like the fuel maps. The timing maps specify a timing offset of between -15 and +15 degrees, where a positive value causes the spark to fire earlier in the engine cycle, and a negative value causes a later spark. Thus, positive values increase the engine's timing advance, and negative values decrease the engine's timing advance.

7.10.3 Gear-Based Fuel and Timing

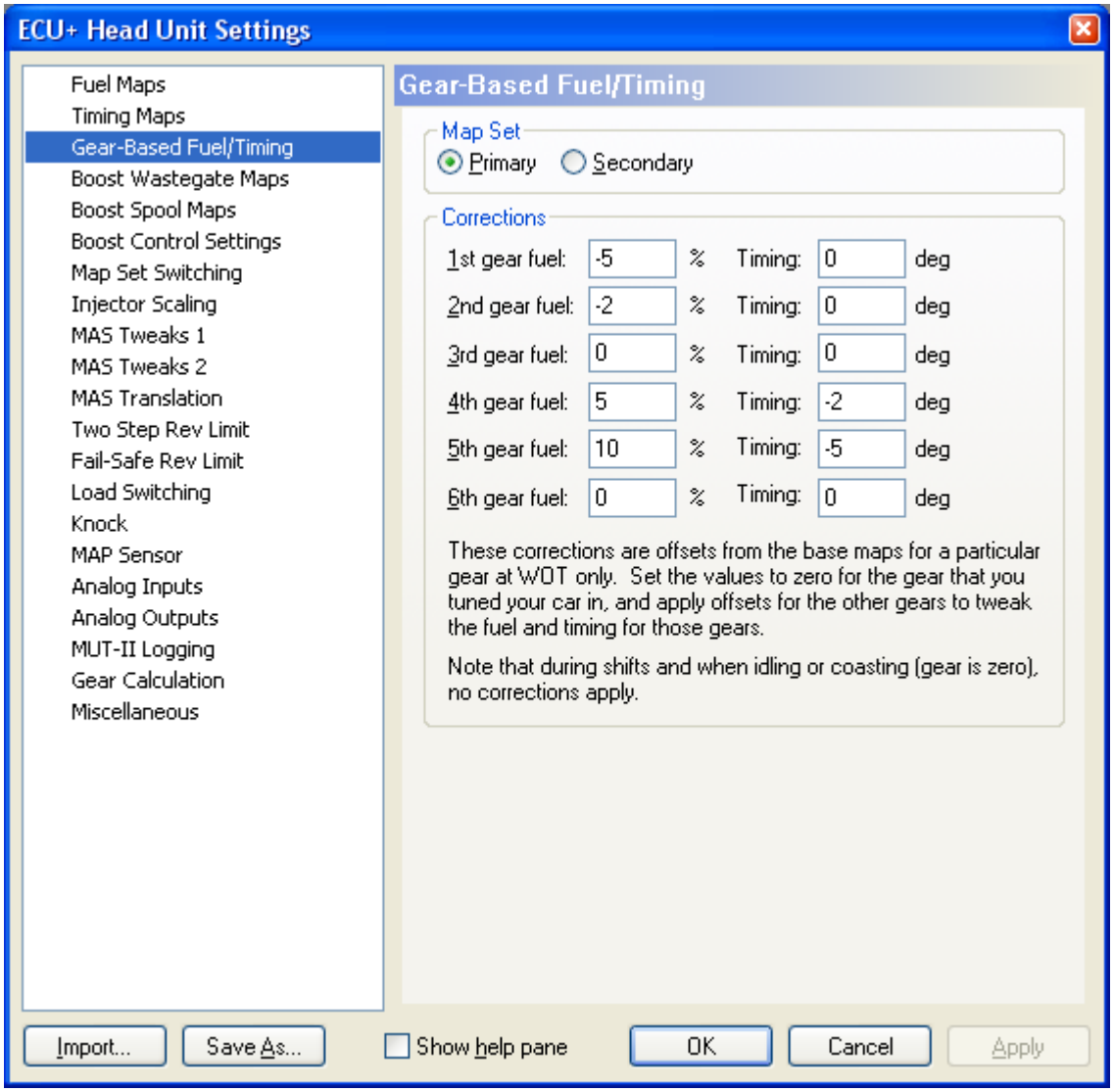


Illustration 74 - The map set switching dialog

In addition to the fuel and timing maps, the ECU+ includes the ability to tweak your fuel and timing on a per-gear basis with the gear-based fuel and timing dialog. These are offsets from the values in the fuel and timing maps, and apply at WOT only. If the ECU+ can't figure out what gear the car is in (as during shifts), no gear offset will apply.

To use these, tune your vehicle in a particular gear (3rd usually) for optimum performance. Then set the offsets here if the car needs a slightly richer or leaner air/fuel ratio in the other gears. Timing offsets can also be done, and is often applied to control knock.

As with the fuel and timing maps, two sets of gear-based settings are available here to be used when the primary and alternate maps are in play.

7.10.4 Boost Wastegate Maps

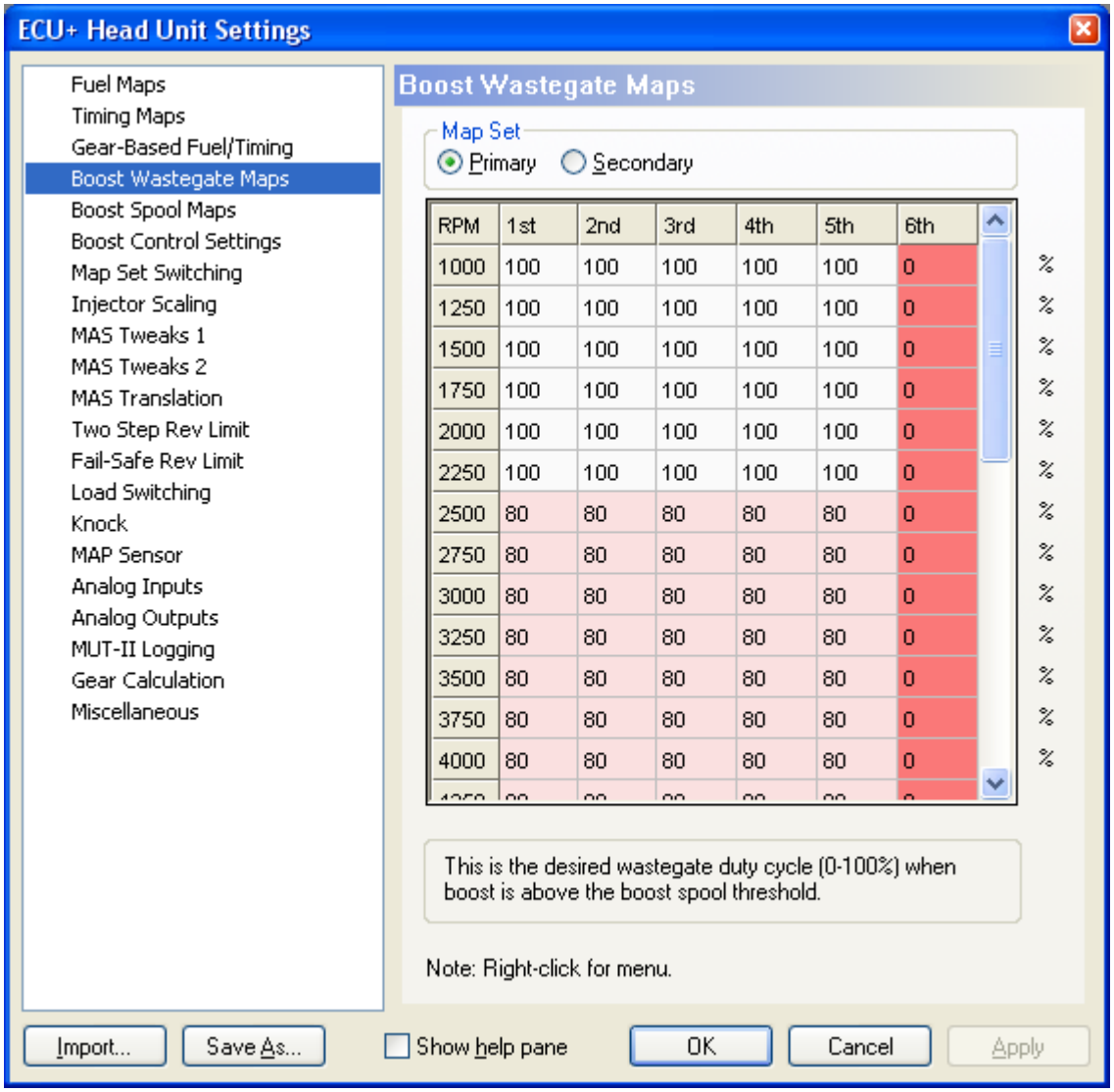


Illustration 75 - The boost wastegate maps dialog

The boost wastegate maps are the primary method of configuring the ECU+ for boost control. The ECU+ controls boost by varying the duty cycle (time on divided by total time) of a solenoid (the boost control solenoid, or BCS) that controls your turbocharger's wastegate. This table specifies, on a per-gear and RPM basis, what wastegate duty cycle the ECU+ should use. Values range from 0% (minimum boost, whose value is set by the spring inside the turbocharger's wastegate actuator) to 100% (maximum boost).

As with the fuel maps, there are two sets of boost wastegate maps, selected at run time by the map set switching controls.

7.10.5 Boost Spool Maps

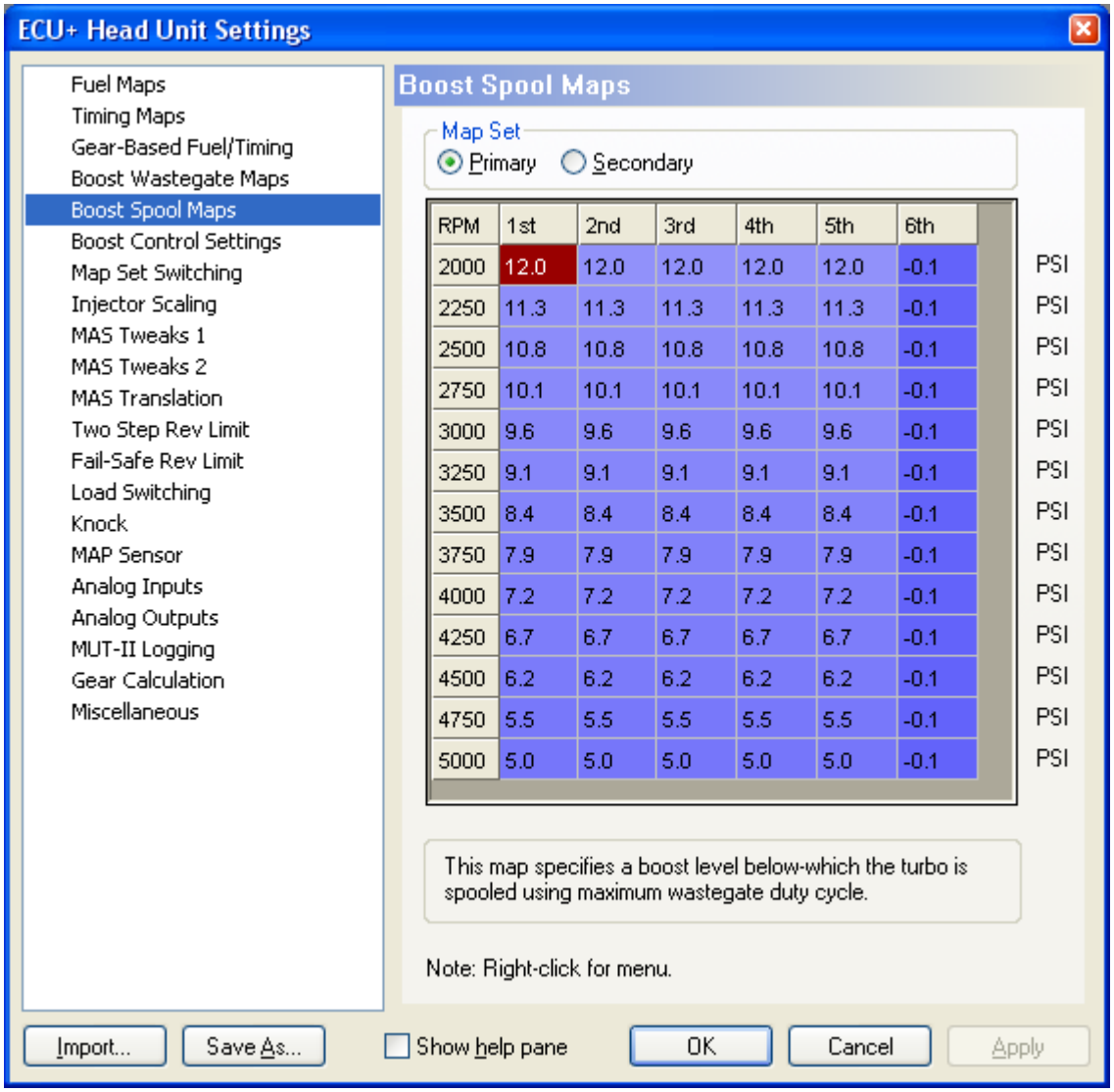


Illustration 76 - The boost spool maps dialog

The boost spool maps allow the ECU+ to spool your turbo very quickly when boost is below a particular threshold value. This table sets a threshold boost level for each gear and RPM. When the ECU+'s measured boost level is below the spool threshold, the ECU+ will use the maximum duty cycle rather than the values in the boost wastegate maps. This causes the ECU+ to spool the turbo as quickly as possible below the spool threshold.

7.10.1 Boost Settings

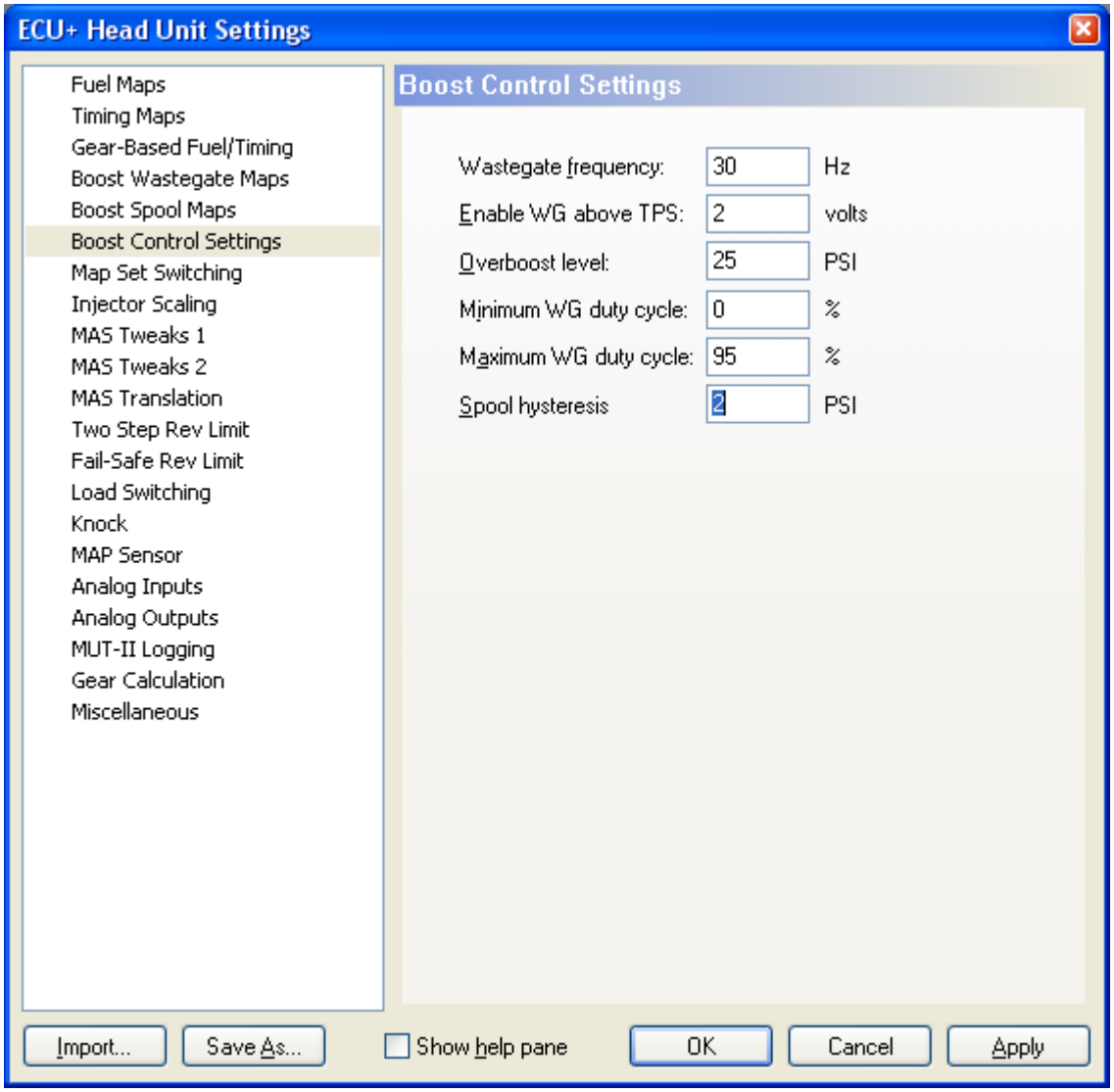


Illustration 77 - The boost settings dialog

The boost settings dialog allows you to tweak some parameters associated with boost control.

- *Wastegate frequency* – The ECU+ cycles the wastegate solenoid on and off based on a duty cycle value. (Duty Cycle is the ratio of time on to time on + time off). This parameter sets the frequency that the wastegate is cycled. Most solenoids use a range of 25-50 Hz. It's rarely necessary to change this value from the default of 30 Hz.
- *Enable WG above TPS* – The ECU+ tries to prolong the life of the wastegate solenoid by not driving it at all in very low boost situations. This parameter controls the minimum throttle position voltage needed before the ECU+ enables boost control. Below this throttle level, the ECU+ drives the wastegate solenoid at the minimum duty cycle (yielding minimum boost). Above this threshold, boost control is active. Note that minimum boost is rarely 0 PSI; rather, the minimum boost level is set by the spring pressure of your wastegate actuator.
- *Overboost level* – This sets the maximum boost level that will be tolerated on your vehicle. Above this boost level, the ECU+ boost control system will try to limit the boost by driving the wastegate solenoid at a minimum duty cycle.
- *Minimum WG duty cycle* and *Maximum WG duty cycle* – These parameters set the minimum

and maximum wastegate solenoid duty cycle that the ECU+ will use when above the overboost threshold and when spooling the turbo, respectively. In general, these should be set to 5% and 95%, as some wastegate solenoids tend to "stick" at high or low duty cycles.

- *Spool Hysteresis* - This is a boost level used for "hysteresis" when switching from not spooling to spooling. When the ECU+ switches from not spooling to spooling, the boost level must drop this much below the value in the Boost Spool Maps before switching back to spooling. A value of 2 PSI usually works well here.

7.10.2 Map Set Switching

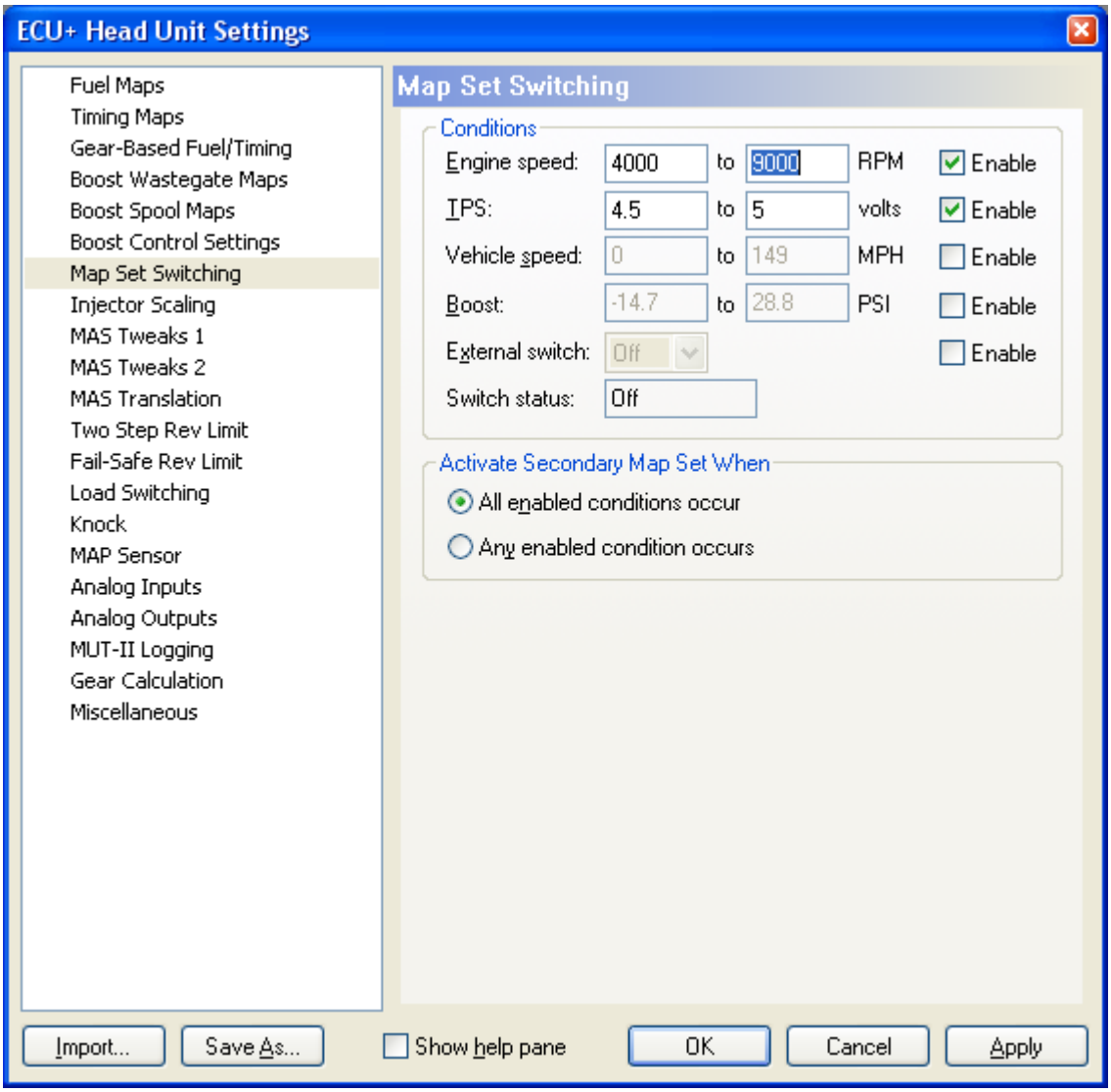


Illustration 78 - The map set switching dialog

As mentioned in the fuel maps dialog, the ECU+ holds two sets of fuel, timing and boost maps – a "Primary" set and a "Secondary" set. The map set switching dialog allows you to configure under what conditions the secondary map set will be used. The ECU+ looks at engine RPM, throttle position, vehicle speed, boost, and the value of the external switch 50 times a second to determine when to switch to the alternate maps, and you can enable and disable any of these selectors with the *Enable* checkboxes.

With the appropriate values enabled, you'll need to select what combination of the enabled values are

used when map switching. Two options are available:

- *All enabled conditions occur* – this is the AND function. Enabled condition 1, and enabled condition 2, and...
- *Any enabled condition occurs* – this is the OR function Enabled condition 1, or enabled condition 2, or...

One of the conditions used is the position of the ECU+'s external switch input. The external switch is an external 5 volt input to the ECU+ head unit that can be wired up to a number of sources (wiring this up is covered in the section The External Switch Input on page 26). The map set switching dialog shows the current value of this switch in the *Switch status* field. The switch status value is updated in real time.

7.10.3 Injector Scaling

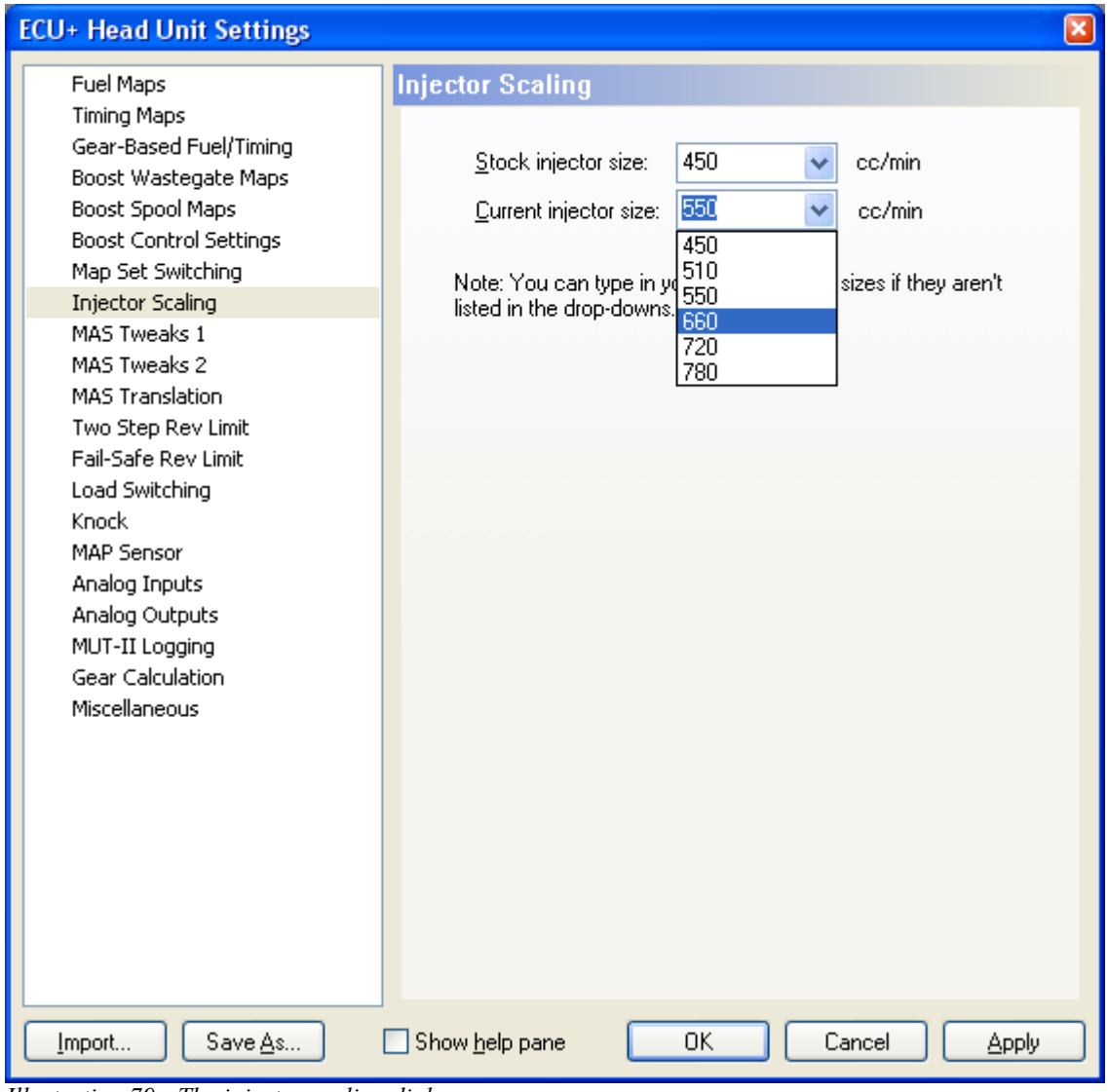


Illustration 79 - The injector scaling dialog

Injector scaling allows the ECU+ to compensate for larger or smaller injectors, while giving you the full -50% to 50% adjustment range in the fuel maps. To use the injector scaling feature, select a new injector size for either the stock or current injectors. From then on, the ECU+ will automatically scale

the MAS output to compensate for the injector ratio defined here.

Values between 391 cc and 780 cc can be entered here, and you can pick the values in from the dropdown list or type them in. Typical stock values are 450 cc on manual-transmission DSMs, 420 cc on automatic-transmission DSMs, and 550 cc on the Evolution VIII and IX.

7.10.4 MAS Tweaks 1

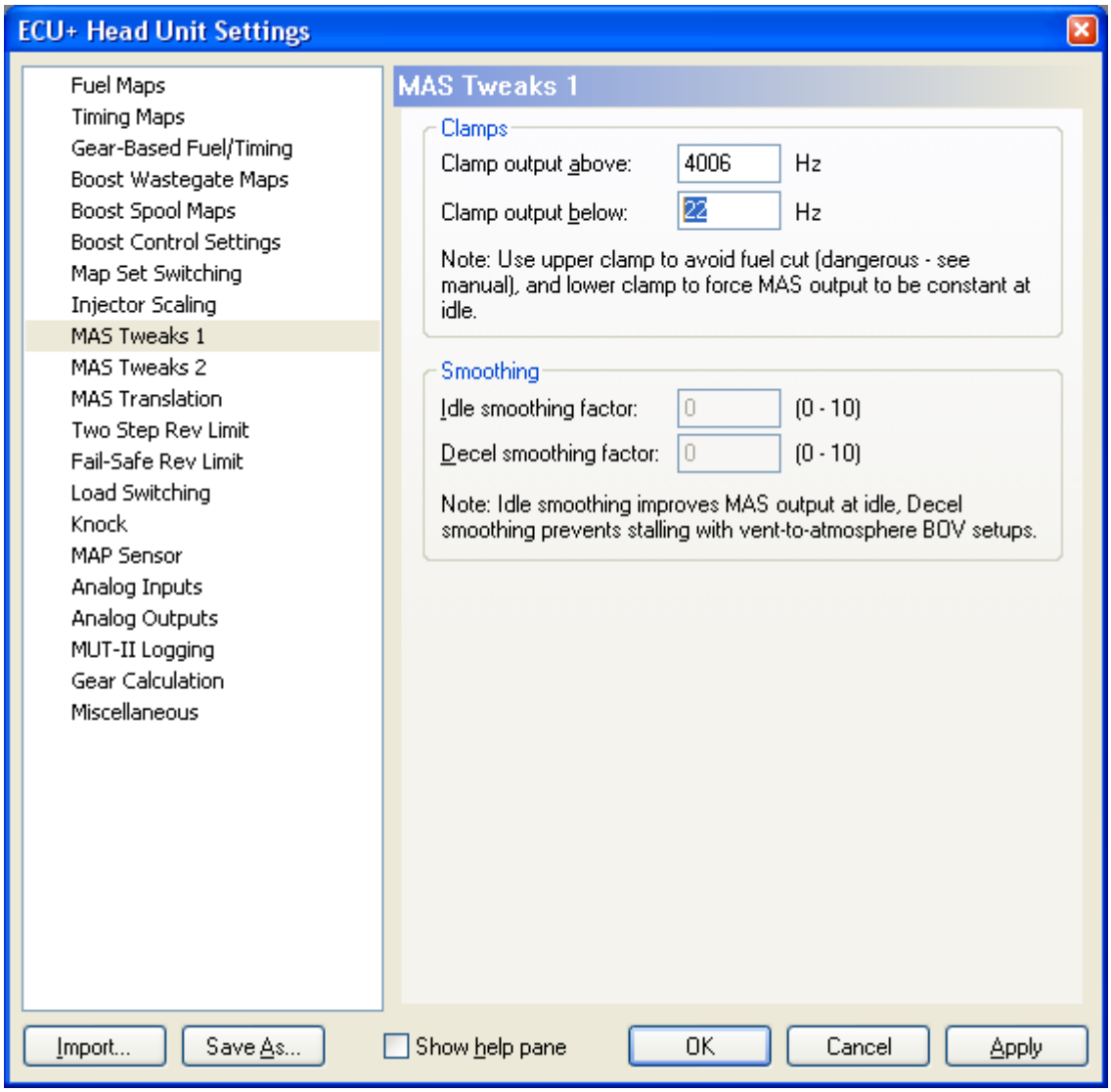


Illustration 80 - The MAS tweaks 1 dialog

As explained previously, the ECU+ modifies the engine's mass air sensor (MAS) signal to make changes to the fuel maps of the engine. This screen allows you to "tweak" the MAS settings in various ways.

- *Clamp output above* - this sets the highest MAS output frequency that the ECU+ will generate on its output. You can use this to limit the output MAS frequency to a given value to avoid fuel cut on your vehicle. You can enter values between 1000 and 4000 Hz here. (The default is 4000 Hz.)
- Some background: the stock DSM and Evolution ECUs look at the frequency of the signal that the MAS generates to determine when to begin "fuel cutting." When the MAS frequency is above a pre-programmed limit, fuel cut kicks in and the stock ECU turns off the engine's

injectors, thus causing the engine to stall momentarily. As soon as the MAS frequency drops back below the pre-programmed limit, normal engine operation ensues.

- By using the MAS frequency clamp, the ECU+ allows you to keep the MAS frequency under the stock ECU's pre-programmed limit and thus avoid fuel cut.
- To configure this value, make a series of full-throttle runs in which fuel cut occurs, and save the captures. Make a note of the MAS *output* frequency of the ECU+ at the point where fuel cut occurs. Then set the clamp to just below that fuel cut frequency.
- Note: running your car with this upper clamp value activated is dangerous! When the normal ECU+ MAS frequency is limited this way, your engine can easily run lean, which can destroy your engine. This option is rarely used, since most high-horsepower engines use large-enough injectors to make this option unnecessary, but if you need to use this option, do-so only along with a wideband O2 sensor to ensure your car never goes into a dangerous lean condition.
- *Clamp output below* – this option is used to smooth out idle for cars with large injectors or aftermarket cams. The clamp below function sets a lower limit on the MAS frequency that the ECU+ will generate at idle, which can be used to set the ECU+'s MAS output frequency to a fixed value at idle. To set this value, start at 20 Hz or so, and watch the variations in your vehicle's warmed-up idle RPMs. Slowly raise the value here until idle smooths out. You'll want to verify that the front O2 sensor voltage cycles with your new *Clamp below* value. Typical *Clamp below* values are in the 40-60 Hz range.
- *Decel smoothing factor* and *Idle smoothing factor* – these configuration options are obsolete, and are only used with older versions of the ECU+ firmware.

7.10.5 MAS Tweaks 2

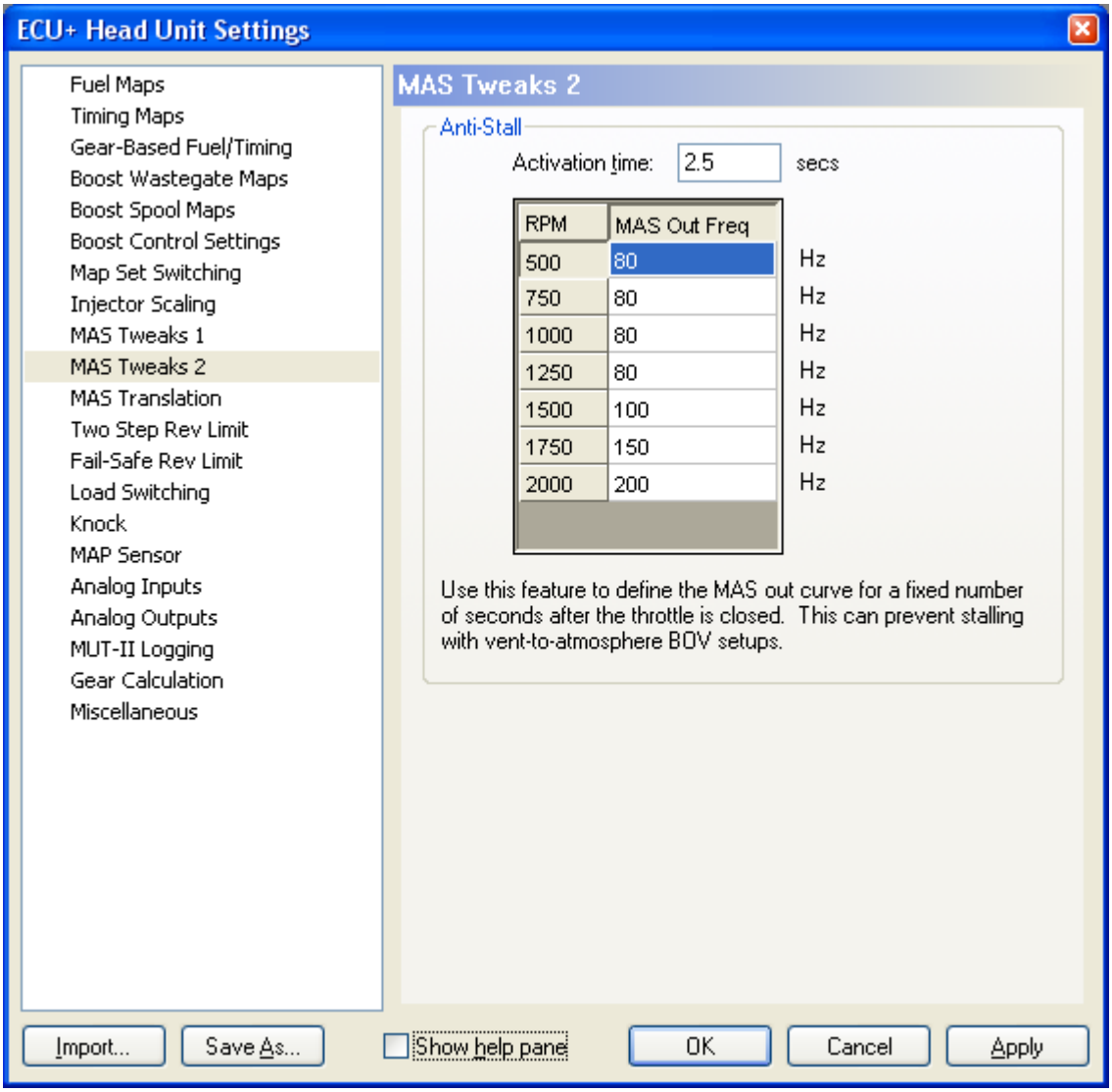


Illustration 81 - The MAS tweaks 2 dialog

The ECU+ modifies the engine's mass air sensor (MAS) signal to make changes to the fuel maps of the engine. This screen allows you to “tweak" the MAS signal to avoid stalling in vehicles with large cams, vent-to-atmosphere BOVs or low vacuum.

Some vehicles have problems when switching from on-throttle to off-throttle, and will stall when coming to a stop. This feature is designed to avoid this condition by temporarily raising the engine RPM. It works like this: when the ECU+ senses that the throttle has gone from open-throttle to closed-throttle, it activates this feature for a few seconds. During that time, the ECU+ sends a table-based MAS frequency to the stock ECU in place of the normal MAS frequency. If the table-based MAS frequency is higher than the normal MAS frequency, RPMs will rise to avoid stalling.

To configure this frequency, consult an ECU+ capture file that has a stall condition in it. From that capture file, determine how long the stall took to happen, and set the ECU+'s Activation time to 1.5 to 2x that time period. Then fill in the MAS out table with MAS out frequency values higher than those seen in the capture file for a given RPM. As needed, raise the MAS out frequencies and activation time to avoid stalling.

7.10.6 MAS Translation

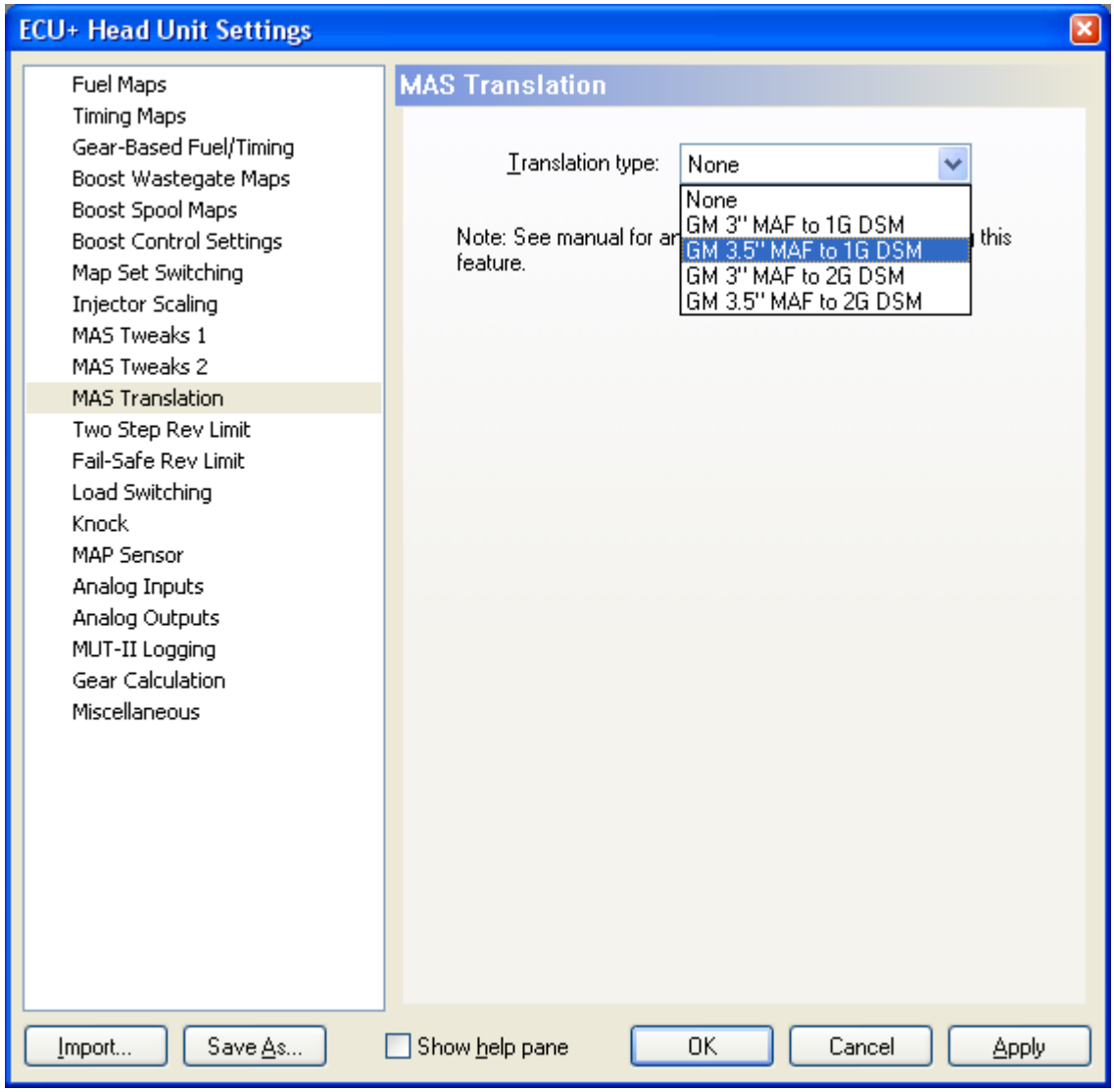


Illustration 82 - The MAS translation dialog

The ECU+ includes the capability to work directly with a GM hotwire-style MAF (mass air flow) sensor. The GM MAFs generally provide less restriction to the flow of air, and can also be used in a "blowthrough" mode in the upper intercooler pipe. When used the GM MAF sensor, the ECU+ "translates" the GM sensor frequency to be compatible with the stock ECU, while still giving you the full range of fuel tuning available with the stock MAS.

On this dialog, select the translation type you want the ECU+ to do.

Using the MAS translation feature requires special wiring and other configuration (beyond this dialog). See the section Using the ECU+ with a GM MAF Meter on page 132 for more information.

7.10.7 Two Step Rev Limit

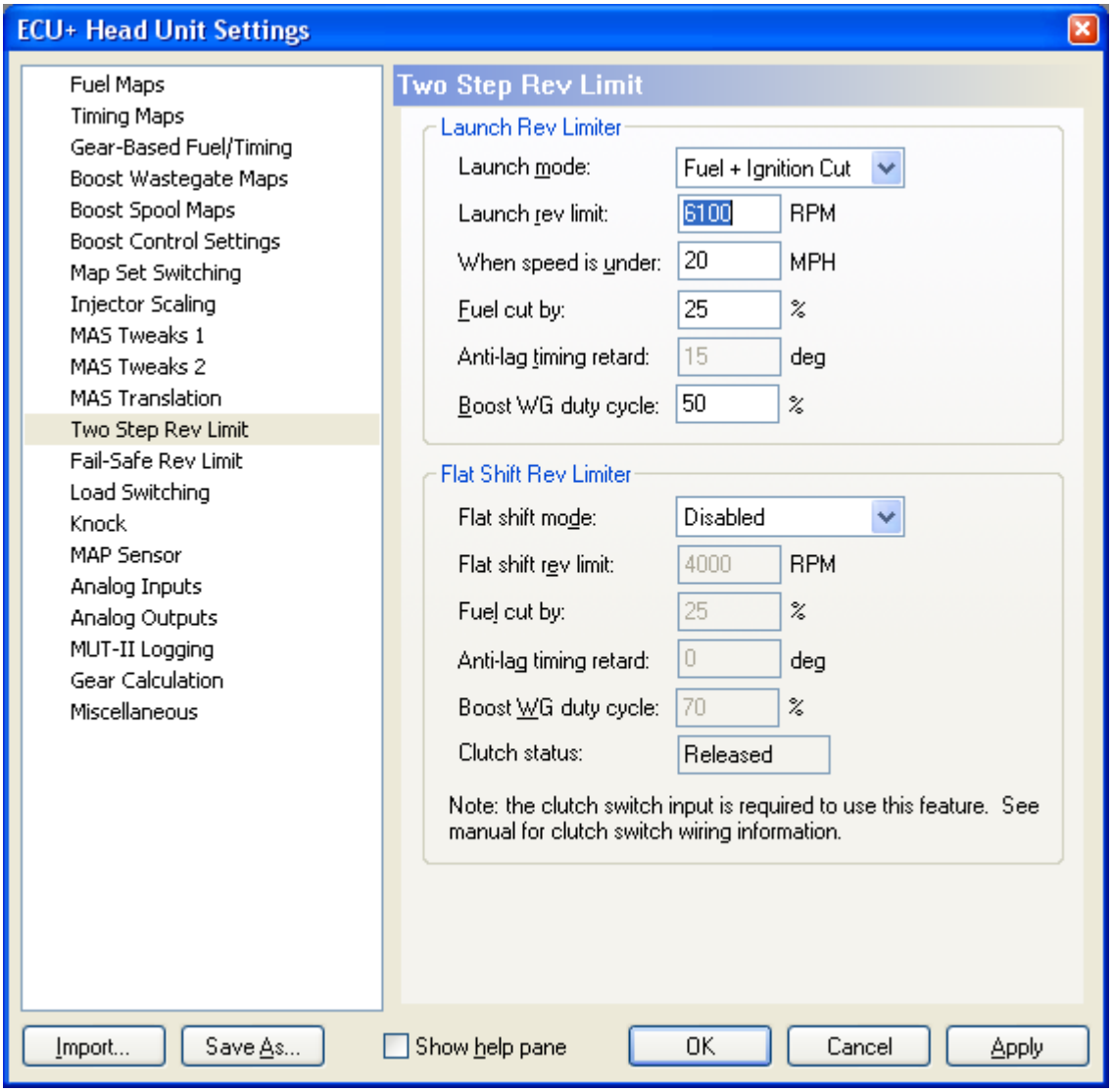


Illustration 83 - The two step rev limit dialog

The ECU+ includes the popular launch control and no-lift shifting features. The two step rev limit dialog configures these two features.

Launch control and no-lift shifting are two variations of a theme. Each is a lower engine rev limit activated under a particular set of circumstances. For launch control, the rev limit is applied when the vehicle is sitting still. No-lift shifting applies the lowered rev limit when the vehicle is moving and the clutch is pressed in. Launch control allows the driver to have a lower rev limit for drag racing, when the car is sitting at "the line." No-lift shifting lowers the rev limit so that the driver can shift more quickly, as the electronics will drop the engine revs down for shifting without lifting off of the throttle.

Both launch control and no-lift shifting are dangerous features that should not be taken lightly. Each is only to be used for *very short* periods of time. Prolonged use of these features (in which the engine revs are artificially limited, possible with dangerous lean conditions) can cause serious knock and ultimately **destroy your engine**. Be *very* careful when using these features.

The two step rev limit dialog configures both the launch control and no-lift shifting functions. You can configure various parameters:

- *Launch and Flat shift mode* – how rev limiting is done. The modes are as follows:

- *Disabled* – no rev limiting is performed. This is the default for both launch control and no-lift shifting. In this mode, the ECU+ does nothing to lower your rev limit.
- *Fuel Cut Only* – the ECU+ will lower your rev limit via fuel cut. In this mode, the ECU+ detects when your RPM exceeds the programmed threshold and cuts fuel by a user-programmable percentage. This mimics your stock ECU rev limiter, which also uses fuel cut to limit RPMs. In this mode, you'll need to experiment with the fuel cut percentage. Set the fuel cut percentage too low, and the fuel cut won't be sufficient to keep the RPMs below your threshold. Set the fuel cut percentage too high, and the engine will bounce off of the rev limit sluggishly. For a given throttle press and RPM threshold, experimentation will find the optimum value. Note that this mode generates a **dangerous lean condition** in your engine, so be careful not to hold the engine in this mode for long periods of time.
- *Ignition Cut Only* – the ECU+ will lower your rev limit via ignition cut. In this mode, the ECU+ turns off the crank and cam sensor signals so that the stock ECU believes the engine has stopped running. This causes the stock ECU to never fire the spark plugs, thus limiting engine RPMs. This mode generally produces a "fast" rev limit, in which the engine bangs against the rev limit quickly. Note that unlike the Fuel Cut Only mode, in this mode the stock ECU will still be firing the injectors, so you'll have unburned fuel dumped into your cylinders during fuel cut.
- *Fuel + Ignition Cut* – this mode combines the two previous techniques. If you set a large fuel cut percentage in this mode, you'll have less unburned fuel in the cylinders during fuel cut. This is the recommended setting.
- *Fuel Cut + Anti-lag* – in this mode, the ECU+ cuts fuel when over the rev limit, just like in the Fuel Cut Only mode. However, this mode also *advances* the engine timing by a user-configurable percentage. This mode is **extremely dangerous** as it attempts to move the engine ignition event past TDC. Moving ignition forward like this can, in some cases, spool turbochargers more quickly than the other modes. It's extremely hard on the turbo and the engine, however, and is not recommended.
- *Launch and Flat shift rev limit* – the RPM at which the feature engages. RPMs between 2500 and 7500 may be entered here. Selecting a launch RPM is a bit of an art – set the launch RPM too low, and the engine will "bog." Set the launch RPM too high, and transmissions, clutches and other drivetrain parts may suffer.

- *Launch speed* – the vehicle speed below which launch control is active. You can enter a speed between 10 and 40 MPH here. In general, you should set this to a low speed so that the launch rev limit will then disengage shortly after launch.
- *Fuel cut percentage* – how much fuel is cut when in one of the fuel cut modes. 0 to 80% fuel cut can be entered here. Generally the higher end of the scale is most useful.
- *Anti-lag timing retard* – how much timing is advanced when using an anti-lag mode. You can enter between 0 and 15% here. Using timing retard is not recommended. See the discussion about about anti-lag.
- *Launch and Flat shift boost WG duty cycle* – the boost wastegate duty cycle to use when launch and flat shift are enabled.

Both launch control and no-lift shifting are configured similarly. You select a mode, and then fill in the other fields to configure that mode. Fields that aren't relevant to a given mode will be grayed-out, and thus unavailable.

In addition to the above fields, this screen also includes a near real-time display of your clutch status. Use this display to verify that your clutch switch is wired correctly. The display shows exactly what the

ECU+ silver box “thinks” is going on with your clutch pedal. If this doesn't match your actual clutch pedal usage, then no-lift shifting will not work correctly. Wiring the clutch switch is covered in the section The Clutch Switch Input on page 24.

7.10.8 Fail-Safe Rev Limit

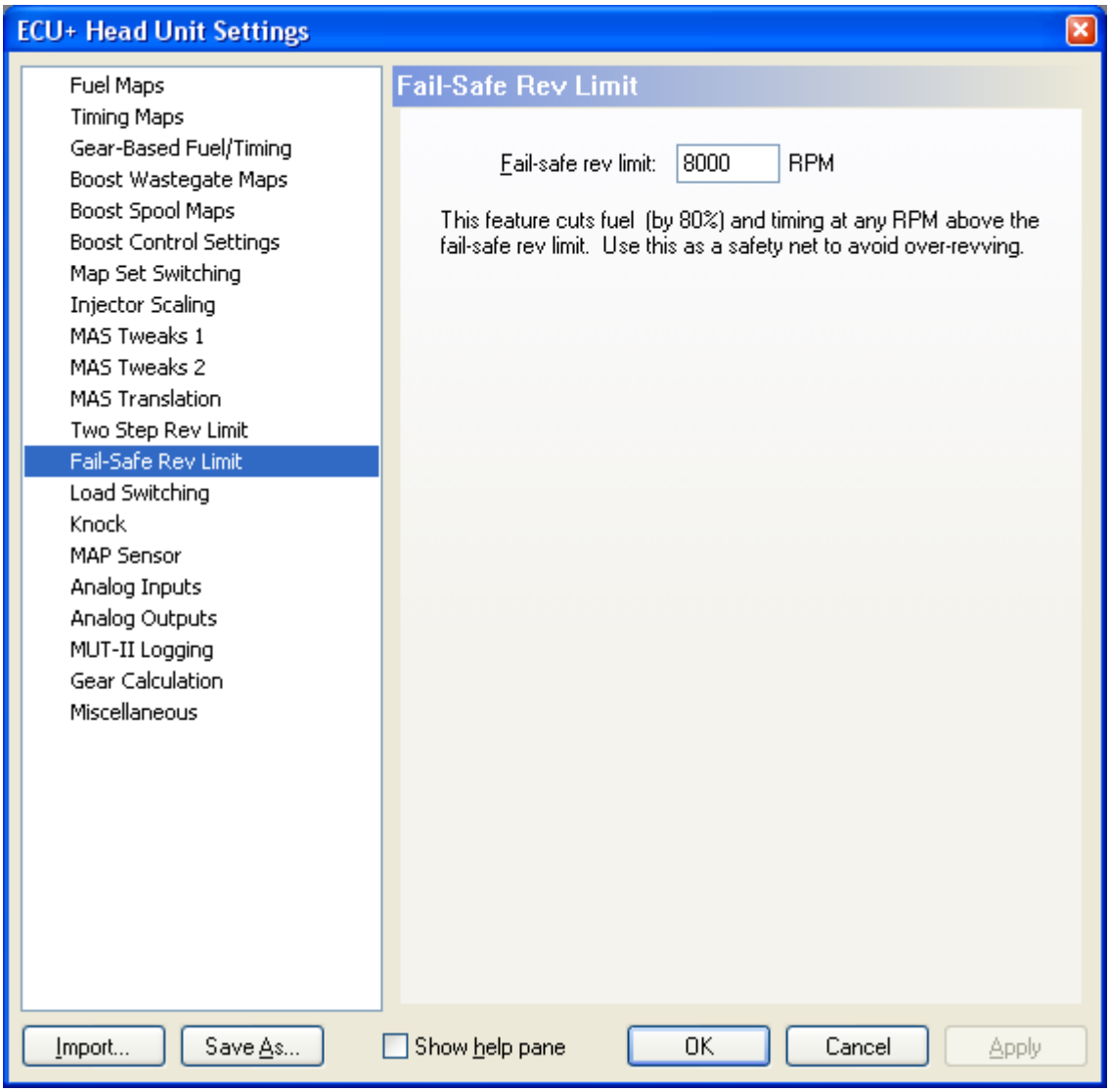


Illustration 84 - The fail-safe rev limit dialog

This page configures the ECU+'s fail-safe rev limit. The purpose of this feature is to allow the overall rev limit of the stock ECU to be lowered below its normal value. Above this fail-safe RPM, the ECU+ cuts fuel (by 80%) and timing.

7.10.9 Load Switching

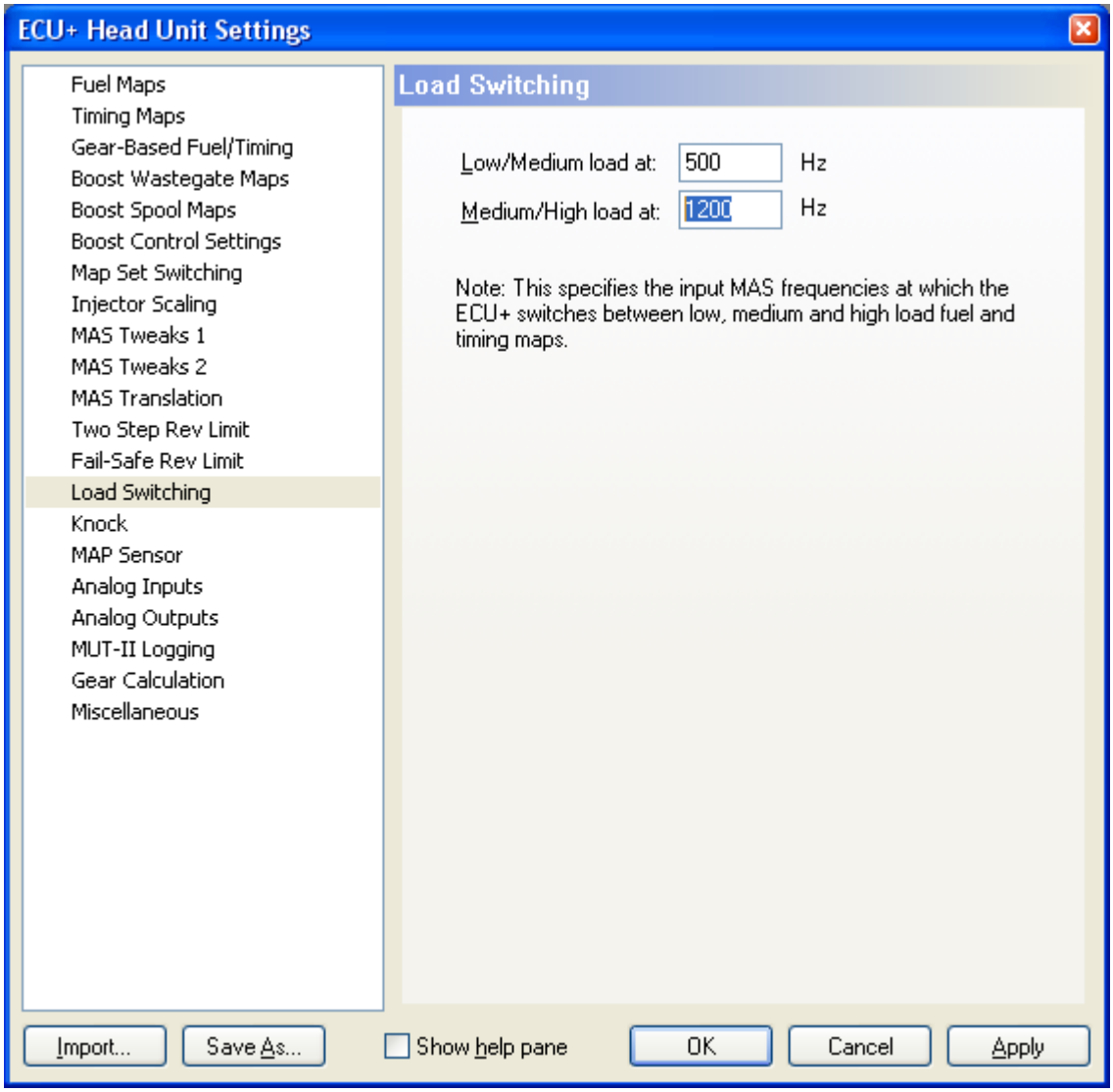


Illustration 85 - The load switching dialog

As shown on page 76, the ECU+ allows you to tune your engine via two 133-cell tables of fuel and timing maps. The maps are indexed by a combination of engine RPM and "load." There are four load values defined: low, medium, high and WOT (idle is a special case of low load). The WOT column is used when the throttle is fully depressed – Wide Open Throttle. The low, medium and high load columns are used depending on what *input* MAS frequency that the ECU+ sees. This screen allows you to fine-tune what input MAS frequencies are used as the switchover point between low-to-medium and medium-to-high loads. Enter new frequencies here to fine-tune when the ECU+ switches between its load maps. If you're not sure, use the default values of 500 Hz (low to medium) and 1000 Hz (medium to high).

7.10.10 Knock

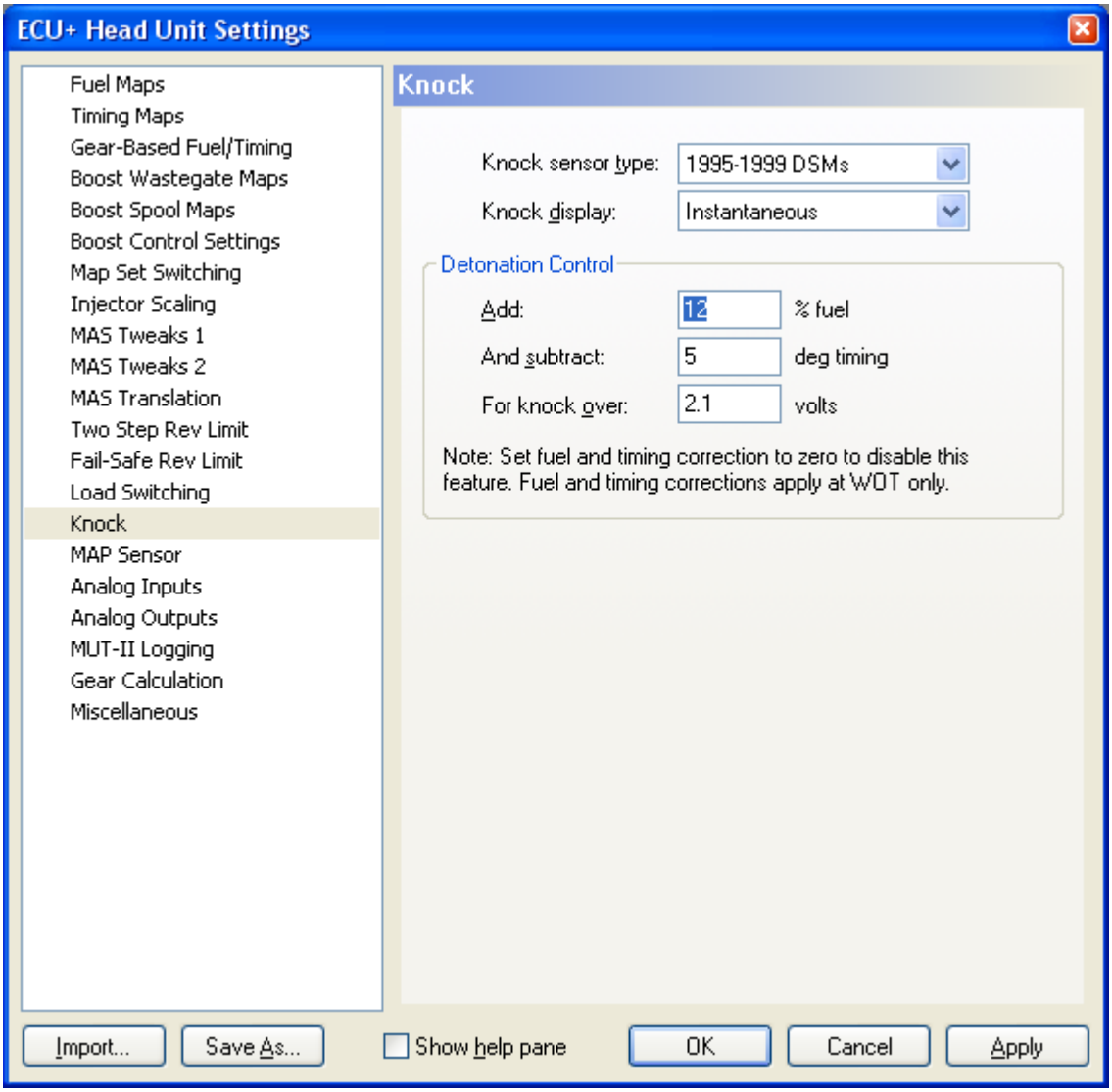


Illustration 86 - The knock dialog

The ECU+ is able to monitor the knock sensor signal on your car and read out a voltage corresponding to how much the engine is "knocking." Knock is very harmful to the motor, and should be avoided at all costs. This screen allows you to configure various knock-related parameters:

- *Knock sensor type* – select what type of knock sensor your car has. In general, this should match your car type.
- *Knock display* – when the ECU+ records and displays knock, it can show the instantaneous knock "events" (knock will appear as a series of spikes on the knock view), or a smoothed-out version of these, where the ECU+ grabs the most recent instantaneous peak and holds it for a short period of time. This setting selects how you'd like to display the knock signal. Instantaneous is the default (and recommended) setting.
- *Detonation control* – when the ECU+ sees a large knock signal, it can help you protect your engine against detonation by automatically increasing fuel and retarding timing for a second or two. The detonation control parameters configure this. Here, enter a knock voltage above which the ECU+ will work its magic, and a percentage of fuel and a timing retard. Fuel values can range from 0 to 25%, and timing values can range from 0 to 15 degrees. (Note that a

positive timing value here retards timing – this is the opposite of how the timing maps work, where a negative timing value retards timing.)

To use the detonation control feature, you should do some captures to determine what values represent a “normal” knock voltage from your engine (you'll see residual knock voltages from normal engine noise). Then set the knock voltage threshold about 0.20-0.50 volts higher than this normal knock voltage. Note that the ECU+'s knock signal will typically show up as a raggedy voltage that increases with RPM. Real knock is indicated by large spikes in the knock voltage. These generally are easy to find, as they'll be accompanied by the stock ECU pulling back timing.

7.10.11 MAP Sensor

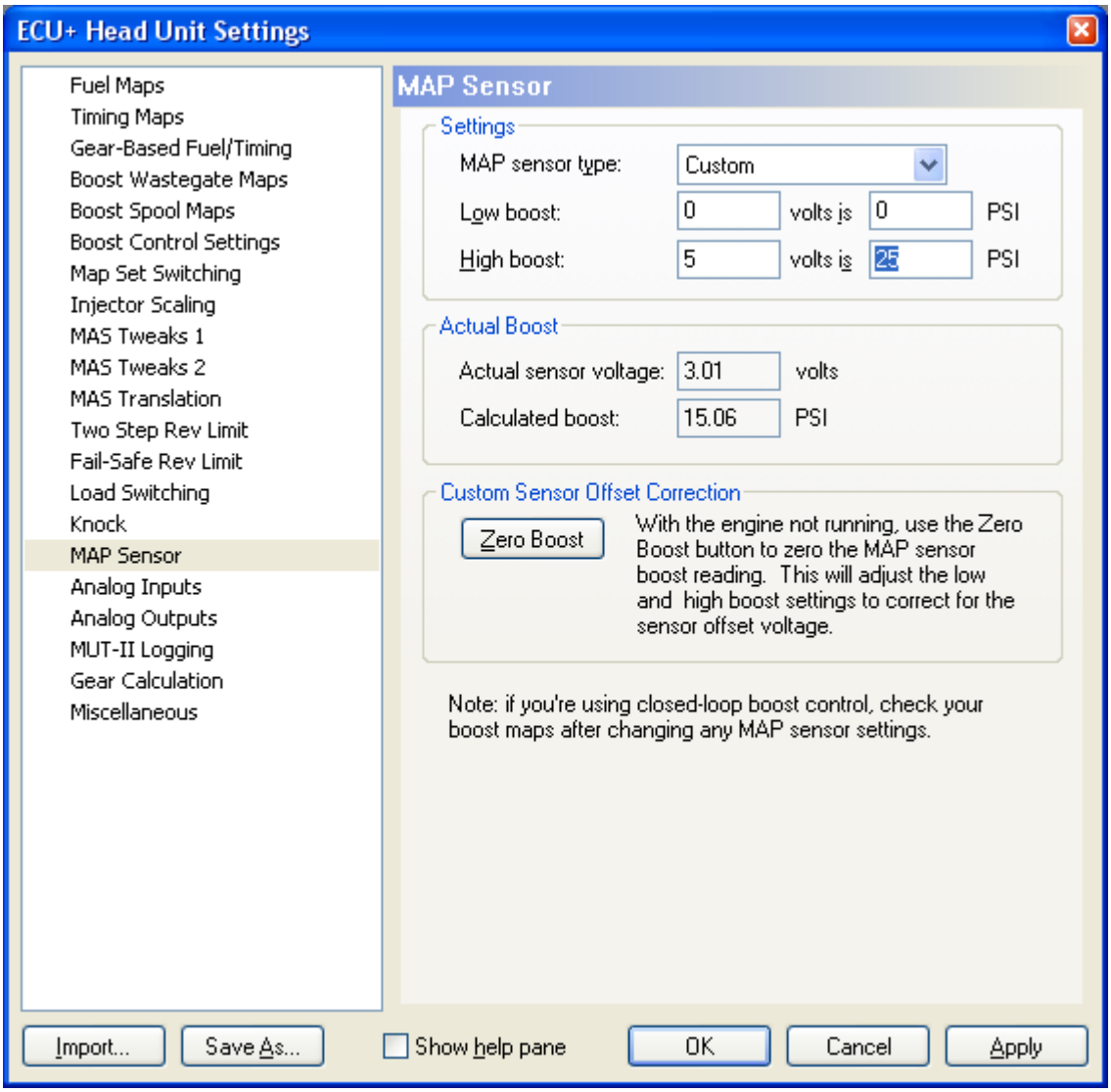


Illustration 87 - The MAP sensor dialog

The ECU+ includes a dedicated analog input that is used with a MAP sensor to allow the ECU+ to log and monitor boost an vacuum levels. Wiring a MAP sensor to the ECU+ is covered in the section Using a MAP Sensor on page 20.

The MAP sensor dialog configures the MAP sensor input to match the type of MAP sensor that's installed on your vehicle. For typical usage, just select your MAP sensor in the *MAP sensor type* field. Five map sensors are currently supported directly:

- The “GM 3-Bar” map sensor.
- The SenSym ASCX30 two-bar map sensor,.
- The AEM 3.5 bar map sensor.
- The AEM 5 bar map sensor.
- The Freescale MPX4250AP map sensor.

If you've installed a unsupported MAP sensor, the ECU+ can still use it with the *Custom* MAP sensor setting. To configure a custom MAP sensor, follow these steps:

1. Install the MAP sensor as per the manufacturer's instructions and wire it to the dedicated ECU+ MAP sensor input. Note that the MAP sensor must not, under any circumstances, generate a voltage of more than 5 volts or less than ground – doing so may damage your ECU+ head unit.
2. Obtain information from the manufacturer of the MAP sensor as to what it's boost-to-voltage curve is. The ECU+ supports MAP sensors with a linear boost-to-voltage curve.
3. Select *Custom* as the MAP sensor type in the MAP sensor dialog. The *Low boost* and *High boost* fields will become active.
4. Enter values in the low and high boost and voltage fields corresponding to your MAP sensor. The low and high boost fields represent the range of boost values that the ECU+ will interpret, so it's best to enter values from the limits of the manufacturer's curve. Typically, MAP sensors will output 0 volts for some boost level and 5 volts for some other boost level. When entering values, use negative boost numbers to represent vacuum.
5. Click *Ok* or *Apply*.

With the custom values used here, your MAP sensor should read out 0 PSI of “boost” when the engine is not running. The MAP sensor dialog includes a display of the current boost level in the *Actual Boost* fields. Use the *Zero Boost* button to correct for any voltage offsets that your MAP sensor has. The *Zero Boost* button updates the low and high boost numbers to set the current MAP sensor boost value to zero.

Note that the *Custom* MAP sensor setting can also be used to improve the calibration of a supported MAP sensor. If a supported MAP sensor's reading doesn't match that of an analog boost gauge, do the following:

1. Make a note of the low and high voltage and boost numbers for your supported MAP sensor.
2. Select *Custom* MAP sensor, and type in the low and high voltage and boost numbers recorded in step 1.
3. Use the *Zero Boost* button to correct for any voltage offsets of your MAP sensor.
4. Tweak the low and high boost values until the ECU+'s MAP sensor reading matches your boost gauge.

7.10.12 Analog Inputs

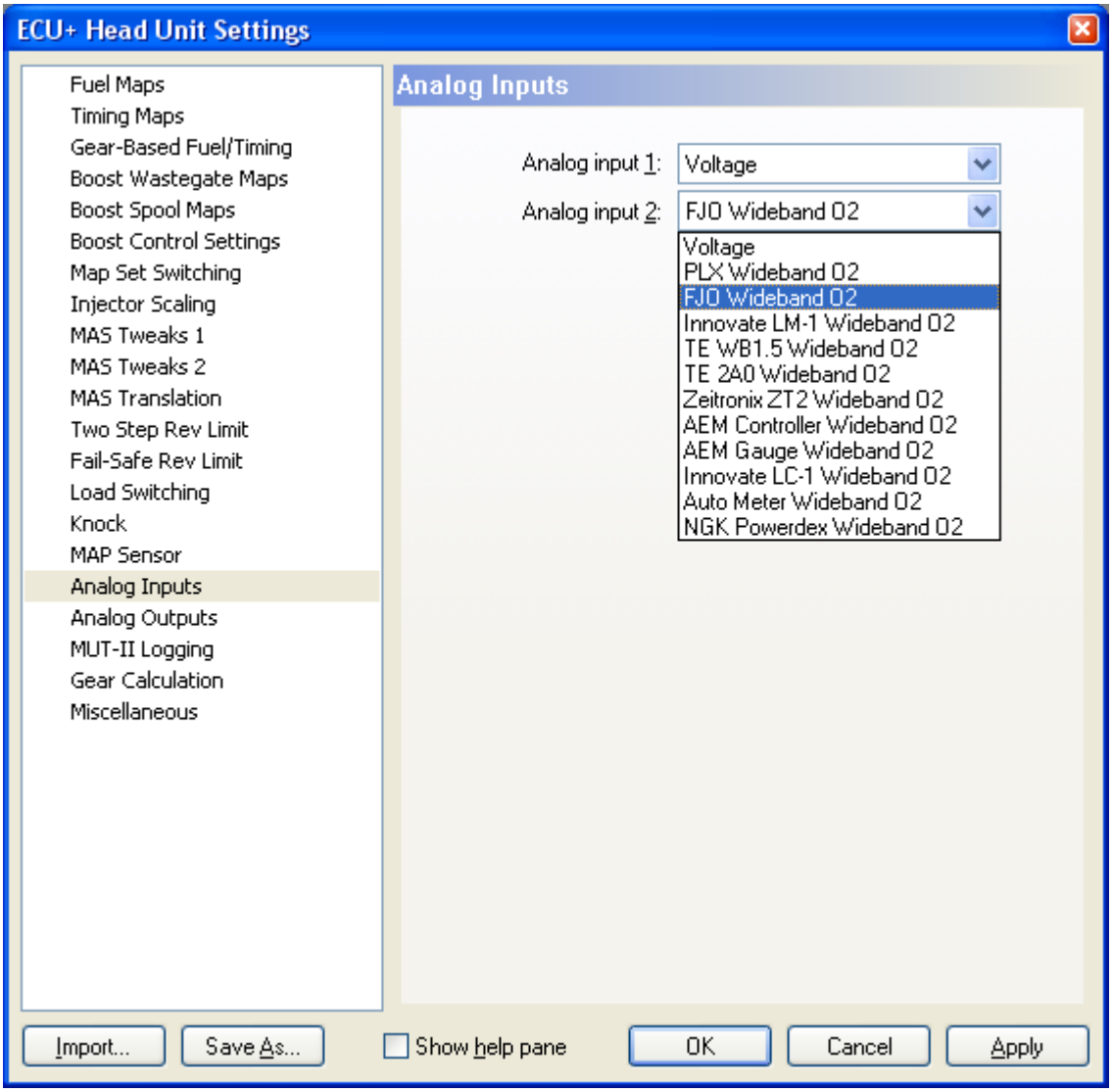


Illustration 88 - The analog inputs dialog

The ECU+ includes three general-purpose 5 volt analog inputs which allow datalogging of any external device that outputs an analog voltage in the range of 0 to 5 volts. Two of the three analog inputs also include calibrations that interpret the analog input in special ways for datalogging. The analog inputs dialog configures what calibrations are used.

Two dropdowns are available:

- *Analog input 1 – a*nalog input 1 can be used as a general-purpose 5 volt input, or it can be hooked to a EGT adapter which converts the signal from an EGT probe into a 5v signal. Select *Voltage* here if analog input 1 was used as a general-purpose input, or select the appropriate EGT adapter type if used with an adapter.
- *Analog input 2* – analog input 2 can be used as a general-purpose 5 volt input, or it can be hooked to a variety of supported wideband O2 sensor kits. Select *Voltage* here if analog input 2 was used as a general-purpose input, or select the wideband sensor kit if it was used with a wideband kit. The ECU+ supports the following wideband O2 sensor kits.
    - PLX Devices M-Series (M-200, M-250, M-300, M-400 and M-500) Wideband Oxygen

Sensor Controllers, with or without the M-Series gauges.

- FJO Racing Wide Band Oxygen Sensor kits using controller CWC0002 or CWC0001.
- Innovate Motorsports LM-1 or LC-1 Air/Fuel Gauge.
- Tech Edge WB02 Lambda Meters (WBo2 version 2A0 and TE-WB Wideband Unit (v1.5)).
- Zeitronix Zt-2 Wideband Air/Fuel Ratio Meter
- AEM Wideband UEGO Controller and Gauge-Type Wideband UEGO Controller.
- Auto Meter Wideband Air and Fuel Ratio Meter.
- NGK Powerdex AFX Air Fuel Monitor.

7.10.13 Analog Outputs

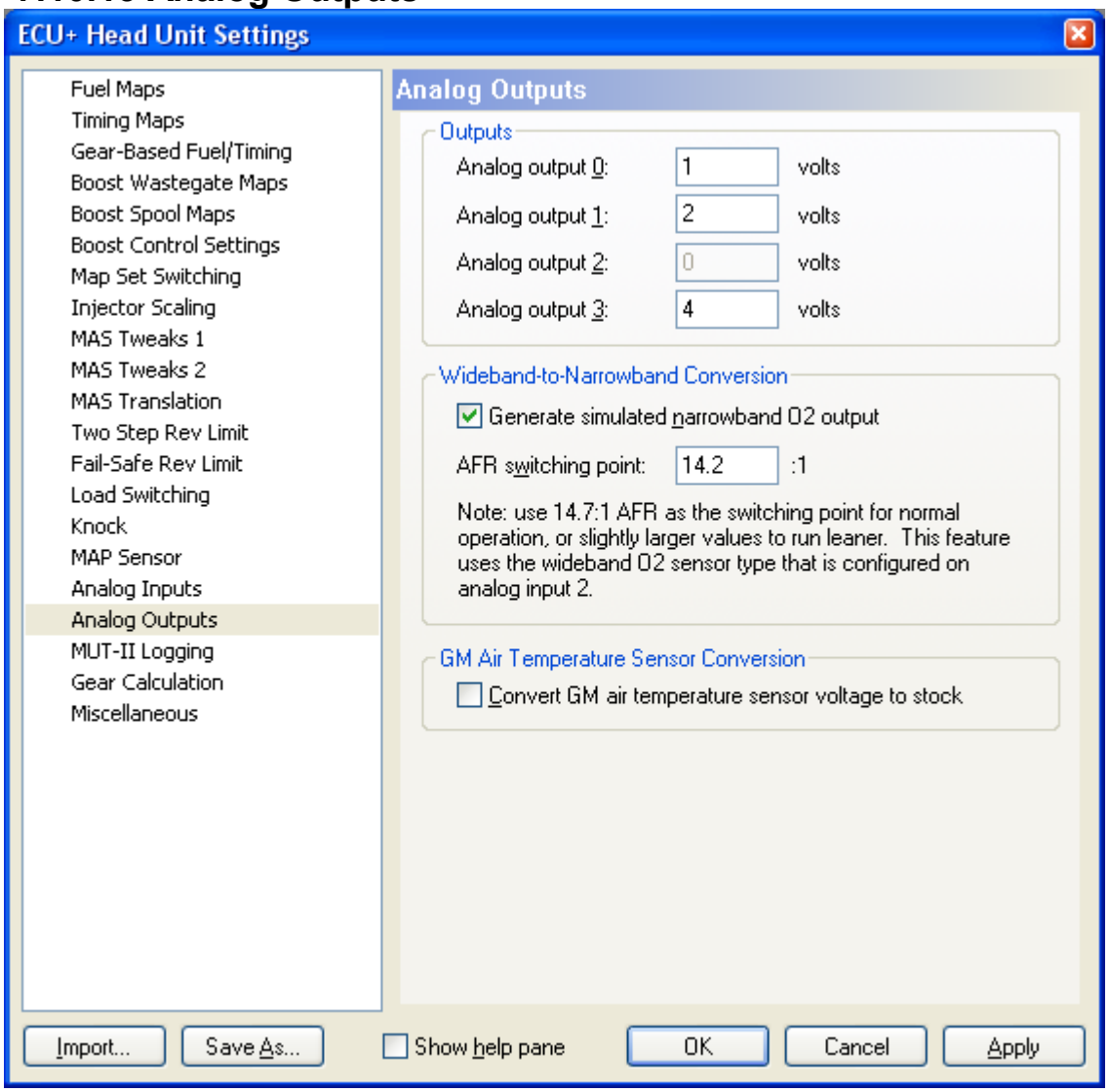


Illustration 89 - The analog outputs dialog

The ECU+ has the ability to generate arbitrary analog voltages on four dedicated analog outputs. This dialog configures what analog voltages you'd like to see on the four outputs. For each of the four outputs, enter a voltage between 0 and 5 volts here. This feature is primarily designed for use with the ECU+'s MAS translation feature, but may be useful to generate other signals as well.

Analog outputs 1 and 2 also includes a special function, as follows:

- Analog output 1 can convert the voltage from a GM air intake temperature sensor to match that of the stock air temperature sensor. To enable this, select the *Convert GM air temperature sensor voltage to stock* checkbox and connect a GM AIT sensor to the first GM AIT sensor input. Analog output 1 can then be connected to the stock ECU air temperature input (Pin 8 on the first-generation DSM ECU, pin 72 on the second-generation DSM and EVO VIII ECUs, and pin 62 on the EVO IX ECU).
- Analog output 2 can generate a simulated front O2 sensor signal based on the input voltage from a wideband O2 kit. In this mode, the ECU+ continuously samples the input voltage on the wideband O2 input (analog input 2) and converts that voltage to the voltage that a narrowband O2 would generate for the same air/fuel ratio. To enable this function, simply check the box labeled *Generate simulated narrowband O2 output* and set the *AFR switching point* as desired. The *AFR switching point* determines the center, or pivot air/fuel ratio of this function. Narrowband O2 sensors pivot at a 14.7 air/fuel ratio, which cause the stock ECU to nominally run at this ratio at idle or low load cruising. By setting this value slightly higher (say, 14.9 AFR), your vehicle will cruise at a slightly leaner air/fuel ratio, potentially saving gas. Note that if no wideband O2 sensor kit has been selected on the analog inputs dialog, the simulated narrowband output will be disabled. More about this special function is available in the section Using the Simulated Front O2 Sensor Output on page 23.

7.10.14 MUT-II Logging

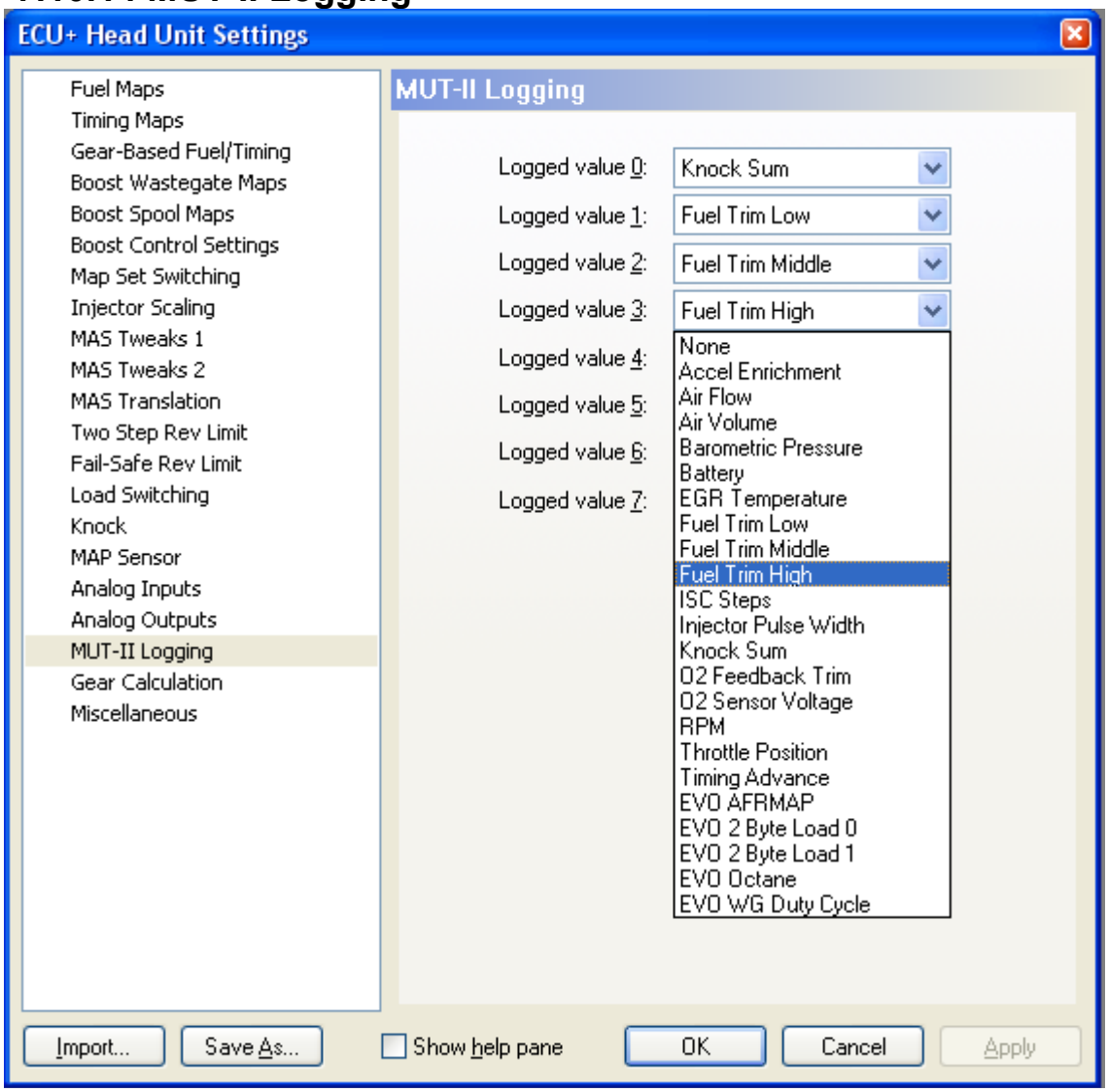


Illustration 90 - The MUT-II logging dialog

In addition to datalogging "physical" connections to the engine sensors (the ECU+'s "native" logging capability), the ECU+ can also log various values that are internal to the stock ECU through a protocol called "MUT-II." MUT-II is supported on all of the vehicles that the ECU+ supports, and allows logging of a fixed set of internal stock ECU values.

The ECU+ can log up to eight MUT-II values at the same time that it logs the physical sensors. The overall MUT-II logging rate is on the order of 50 logged values per second (depending on the type of stock ECU's – the Evolution ECUs are fastest, the first generation DSM ECUs are the slowest), and the ECU+ round-robins between values. That means that if the overall rate is 50 values per second, and you log four different stock ECU values, each value will be change 12.5 times per second. For this reason, it's not recommended that you log more than 3 or 4 MUT-II values at the same time.

The MUT-II values that can be logged are as follows:

- *Accel Enrichment* – the additional fuel percentage that the stock ECU applies when accelerating. A percentage.
- *Air Flow* – the measured MAS frequency as seen by the stock ECU, in Hz. Similar to the

ECU+'s native logging of MAS output frequency, but not as accurate. Note that this is limited to 1603 Hz by the stock ECU.

- *Air Volume* – a calculated air flow number, based on MAS frequency, temperature and barometric pressure.
- *Barometric Pressure* – the barometric pressure of the input air, as a number of bar (1 bar is 14.7 PSI).
- *Battery* – the measured battery voltage, in volts.
- *EGR Temperature* – the exhaust gas recirculation temperature.
- *Fuel Trim Low, Middle and High* – a calculation of how much the stock ECU has to adjust its fuel tables away from the factory defaults to achieve the optimum 14.7 air/fuel ratio at cruising. These are 100 when no adjustment is necessary. Three trims are available here, representing the short, medium and long-term trims, respectively.
- *ISC Steps* – the number of steps of the idle speed control servo. The ISC is a mechanism that passes air around the throttle plate (which is closed at idle) to control idle speed. The ISC increases air flow at idle when under certain conditions, like A/C on and hot starts.
- *Injector Pulse Width* – the injector pulse width, in ms.
- *Knock Sum* – the knock sum is the most useful MUT-II parameter. It is a number that the stock ECU calculates to indicate that the engine is knocking. Small values (under 10 or so), indicate low levels of knock, while high values of knock sum indicate a knocking condition that the stock ECU deals with by retarding timing. This is similar to the ECU+'s knock voltage, but includes various factors that the stock ECU applies to the knock voltage to correct it for known engine noise.
- *O2 Feedback Trim* – another trim value.
- *O2 Sensor Voltage* – the measured voltage on the front O2 sensor, as a number of volts.
- *RPM* – the measured engine RPM. Not as accurate as the physical measurement done by the ECU+'s native logging.
- *Throttle Position* – the throttle position, as a percentage. Note that the ECU+'s native logging shows throttle position as a number of volts, whereas this is a 0 to 100% range.
- *Timing Advance* – the engine timing, as set by the stock ECU. In general, this will not match the ECU+'s native timing advance value, as the ECU+ "fools" the stock ECU by altering the cam and crank sensor signals.
- *EVO AFRMAP* – this logs a "map position" useful when calculating "load" in the ROM editing and map tracing version of ECU+ Win. *Evolution VIII and IX only.*
- *EVO 2 Byte Load* – an alternative to logging the AFRMAP on the Evolution cars, this logs a high resolution load value. There are two 2 Byte Load values, and both must be logged to enable load tracing in the ROM editing and map tracing version of ECU+ Win. *Evolution VIII and IX only, with a modified ROM that supports this parameter.*
- *EVO Octane* – a calculated estimate of the octane, or knock resistance of the type of gas in the vehicle. This ranges from 0-100, and determines whether the stock ECU is using the low or high octane maps. *Evolution VIII and IX only.*
- *EVO WG Duty Cycle* – duty cycle (0-100%) of the boost control wastegate. *Evolution VIII and IX only, with a modified ROM that supports this parameter.*

In addition to the standard values listed above, the ECU+ can also log *any* MUT-II value by using the

ECU+'s custom MUT-II logging functionality, as described in the section MUT-II Logging Configuration on page 47. Once these custom MUT-II commands have been defined, they'll appear in the drop-downs shown above along with the standard MUT-II values.

In general, you'll want to log knock sum, one or two of the trims, and the EVO AFRMAP value on the Evolution VIII or IX cars. The other values are generally not that interesting, or are logged better and faster by the ECU+'s native logging capabilities.

7.10.15 Gear Calculation

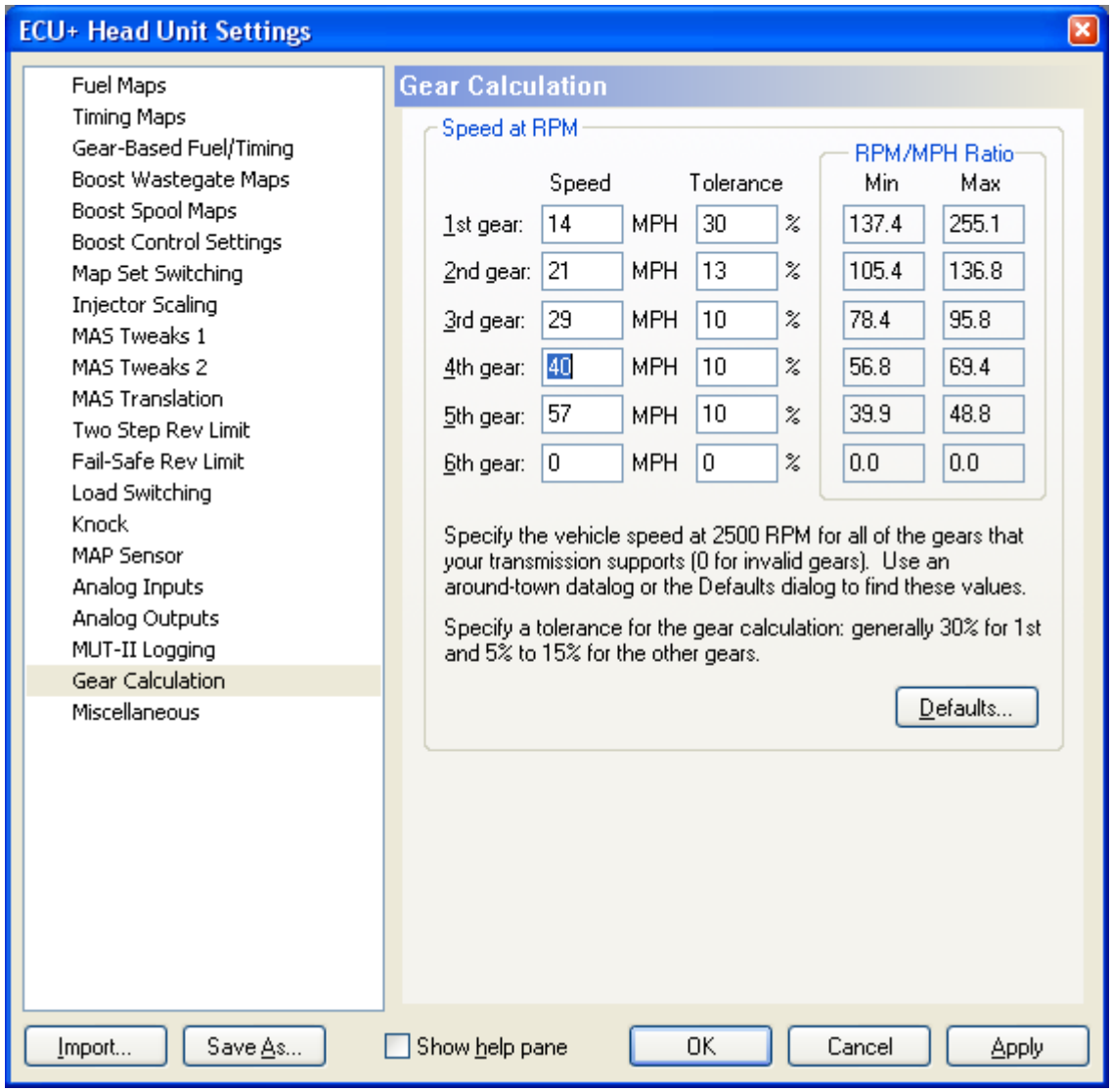


Illustration 91 - The gear calculation dialog

The ECU+ has the ability to datalog and make changes to fuel, timing and boost values as a function of the vehicle's current gear number. The gear calculation dialog allows you to configure the gear ratios used on your car.

To calculate a gear number, the ECU+ divides the current engine RPM by the vehicle speed (in MPH). This ratio is then looked up in a table of .RPM/MPH ratios (one per gear) and if the calculated ratio matches the table ratio (with some tolerance) the ECU+ assumes the associated gear number is what gear the vehicle is using. The gear calculation dialog sets the ratios that the ECU+ will compare against.

For most cars, setting up the ratios is as simple as clicking the *Defaults* button, which displays a list of known transmission types and stock tire sizes for a variety of supported vehicles. This dialog is shown below.

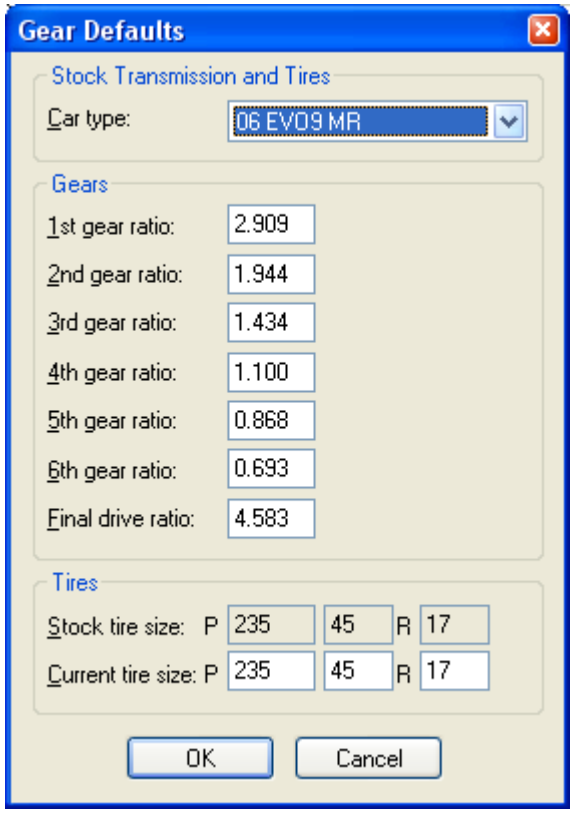


Illustration 92 - The gear defaults dialog

To use this, either accept the defaults for the transmission ratios (just about everyone) and tire sizes (some people) or make changes to match your vehicle, then hit the Ok key. The proper gear ratios and tolerances will be carried back to the gear calculation dialog.

To verify that the gear values are correct, you'll want to do a capture with the custom log value of Gear set up. (More information about setting up custom log values is available in the section Custom Logging Configuration on page 44.) If all is well, the proper values will be logged. If not, follow this procedure:

1. Set up two custom log values for Gear and RPM/speed, as shown below.

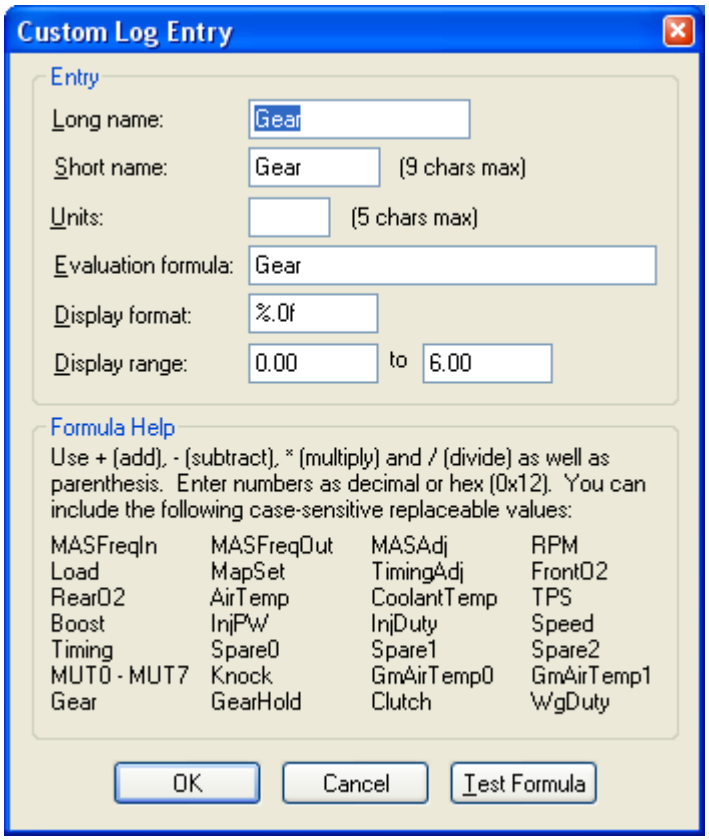


Illustration 93 - Gear custom log entry

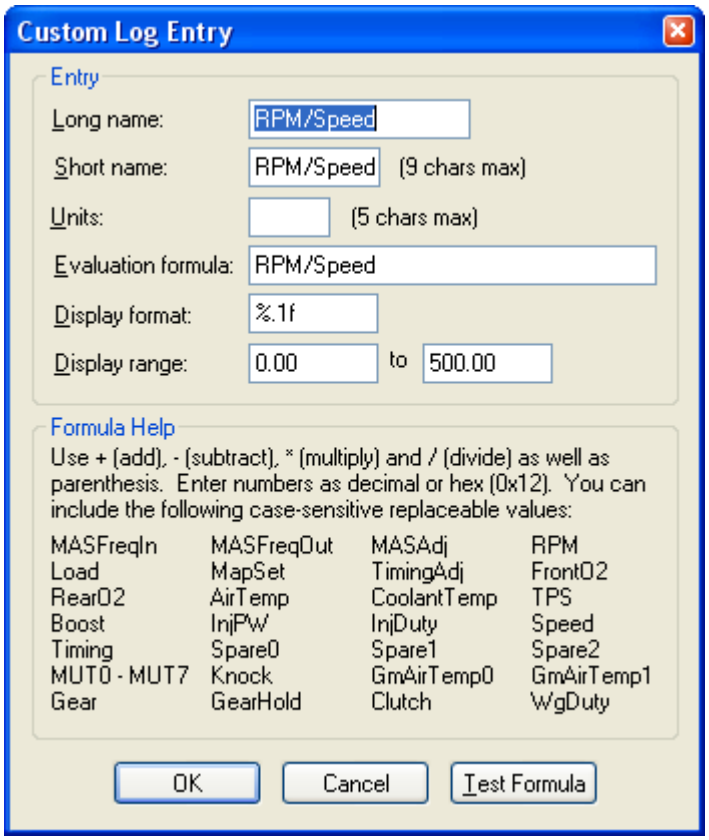


Illustration 94 - RPM/Speed custom log entry

2. Next, do a capture with these values configured. Drive around in all of your vehicle's gears, and try to stay in each gear across a wide range of RPMs (say 2000-5000 RPM). Save this capture.
3. Open up the above capture in the ECU+ Win software and take a look at the views for RPM, Vehicle Speed, Gear and RPM/Speed. You should see something like this:

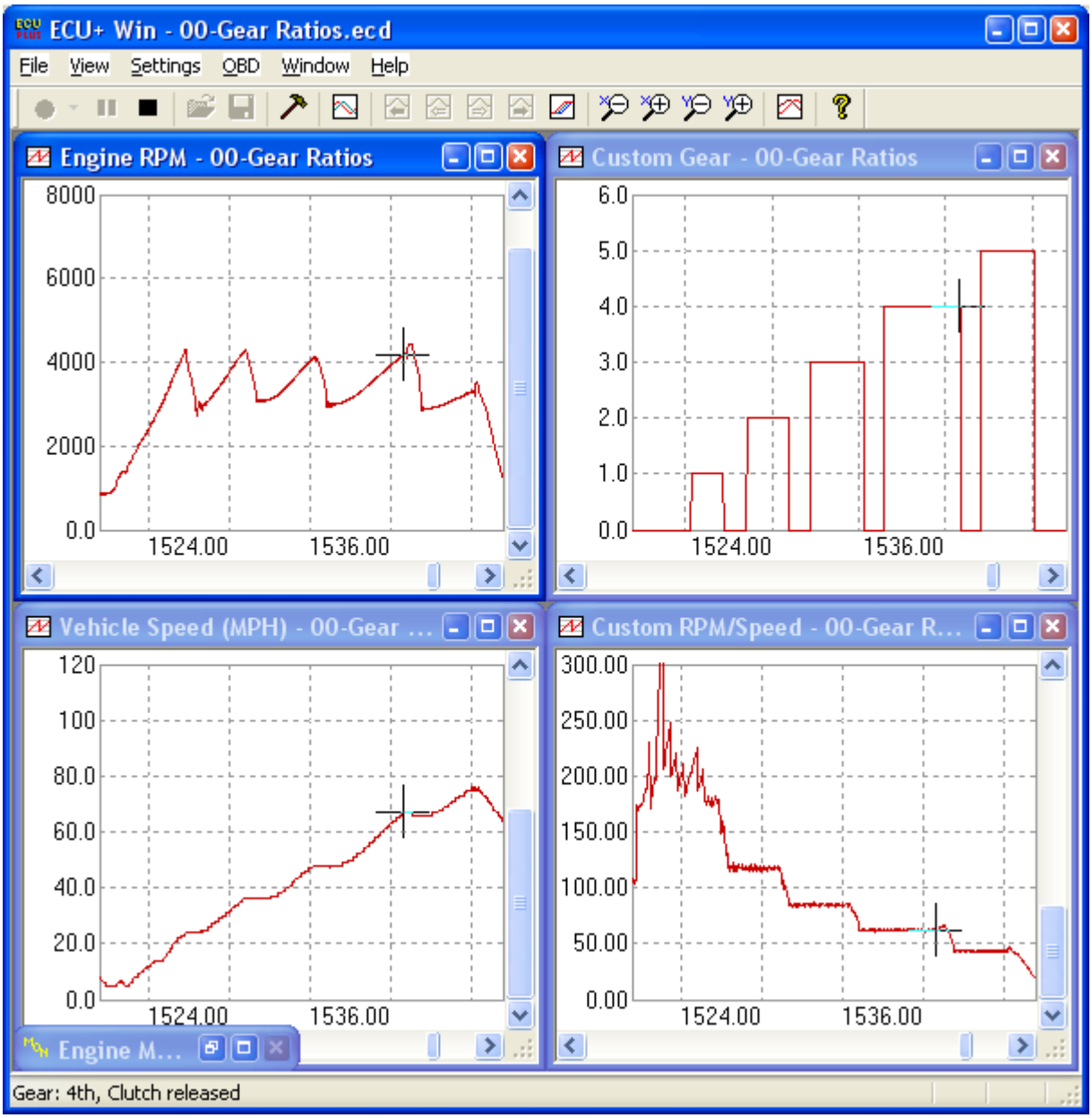


Illustration 95 - Custom gear ratio calculation

4. From looking at the graphs it should be obvious when the vehicle is in various gears. Flat spots in the RPM/Speed graph correspond to when the shift lever is fully engaged. Compare the calculated RPM/Speed values with the RPM/MPH min and max ratios shown in the gear calculation dialog and adjust the values as needed. You'll generally find that the tolerances for various gears will be different, and that 1st will need the largest tolerance.
5. You'll find that for certain gears, the ECU+ software will miscalculate the proper gear number. For those gears, simply adjust the speed and tolerance fields so that the the RPM/Speed graph values match the ratios in the dialog.

For most vehicles, this procedure isn't necessary, but vehicles with unusual tire sizes or gear ratios may require this tweaking.

7.10.16 Miscellaneous Settings

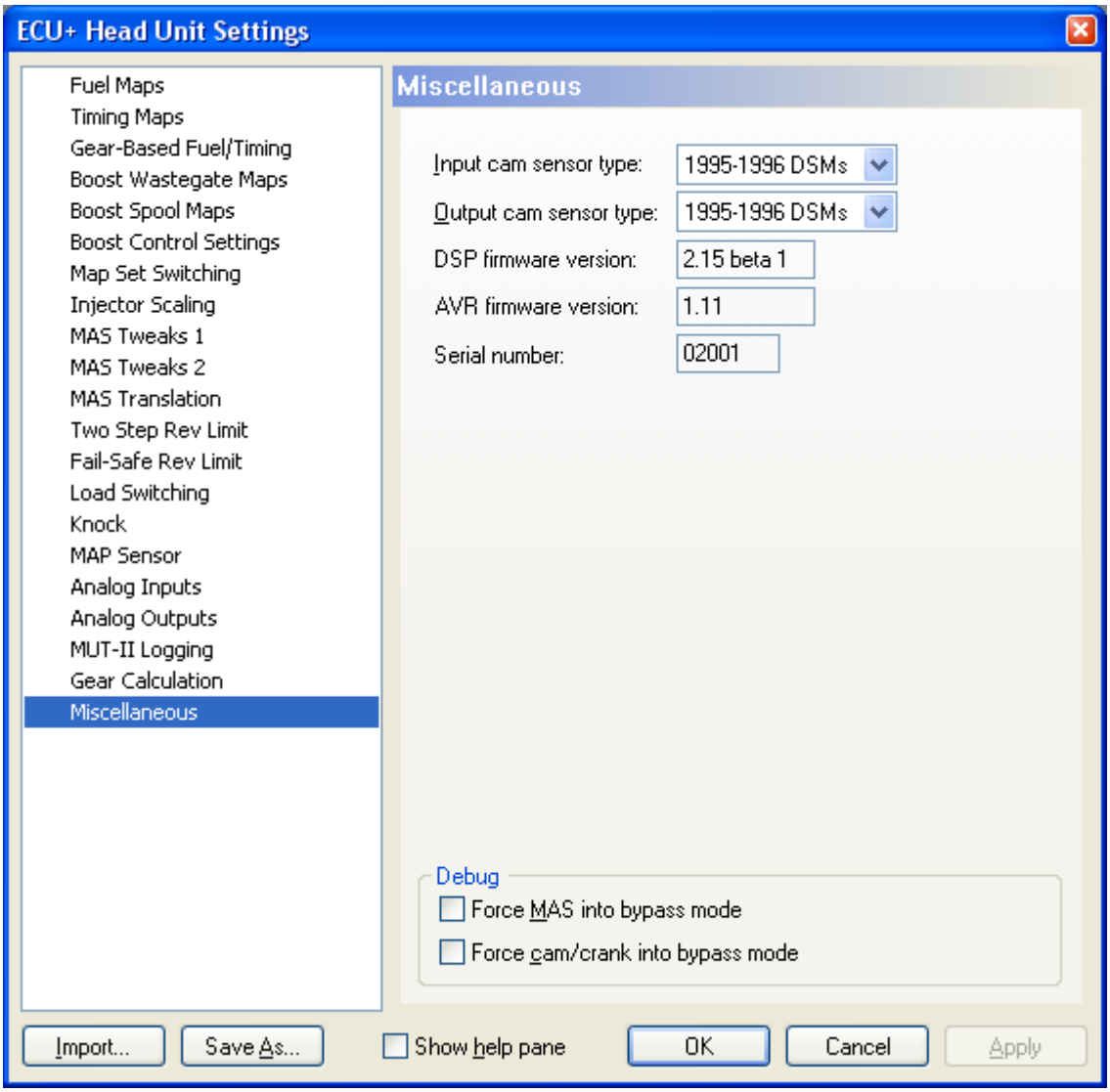


Illustration 96 - The miscellaneous settings dialog

The miscellaneous settings dialog lets you configure some random values in the ECU+ head unit. These are:

- *Input cam sensor type* - this sets the type of cam sensor used on your car. Click the dropdown button next to the type to set a new value.
- *Output cam sensor type* - this sets the type of cam sensor that the ECU+ head unit should generate. As with the *Input cam sensor type*, click the dropdown arrow to set a new value. Both the input and output cam sensor types should generally match your car - this setting is intended for those people using a different cam sensor than stock on their car. This is most useful for 2G DSM cars with a "6 bolt" motor configuration that has a 1G cam sensor. For that setup, configure the input cam sensor for 1990-1994 DSMs, and the output cam sensor to match your stock ECU.
- *DSP firmware version* – the ECU+ includes two microprocessors inside the silver box – a "DSP" microprocessor and an "AVR" microprocessor. This displays the version of the DSP firmware running in the ECU+ head unit. Use this when submitting bug reports.

- *AVR firmware version* – this is the version of the AVR microprocessor firmware.
- *Serial number* – this displays the serial number of your ECU+ head unit.
- *Force MAS into bypass mode* - This is a debugging option that forces the ECU+ to track your MAS sensor signal with no changes. This effectively causes your car's MAS sensor signal to pass transparently through the ECU+ head unit.
- *Force cam/crank into bypass mode* - Similar to the MAS Bypass option, this forces your car's cam and crank sensor signals to pass transparently through the ECU+ head unit. Note that neither the MAS Bypass nor the Cam/Crank Bypass check-boxes are remembered by the ECU+ head unit - these are both switched off when the head unit is powered down, and have to be re-enabled each time you start your car.

7.11 Saving and Loading Head Unit Settings

The ECU+ can "hold" one complete set of the head unit settings. Sometimes, it's desirable to maintain multiple sets of head unit settings (for example, a set for pump gas, a set for race gas and a set for nitrous). To enable this, the ECU+ Win software can load and save a set of head unit settings to a file on your laptop's hard disk.

On any of the head unit dialogs there are two buttons, *Import* and *Save As*. Use the *Import* button to import a set of head unit settings, or the *Save As* button to save your settings to disk. In each case, you'll be prompted for a filename to import from or save to, and when importing, you'll have the option of only importing a portion of the saved data.

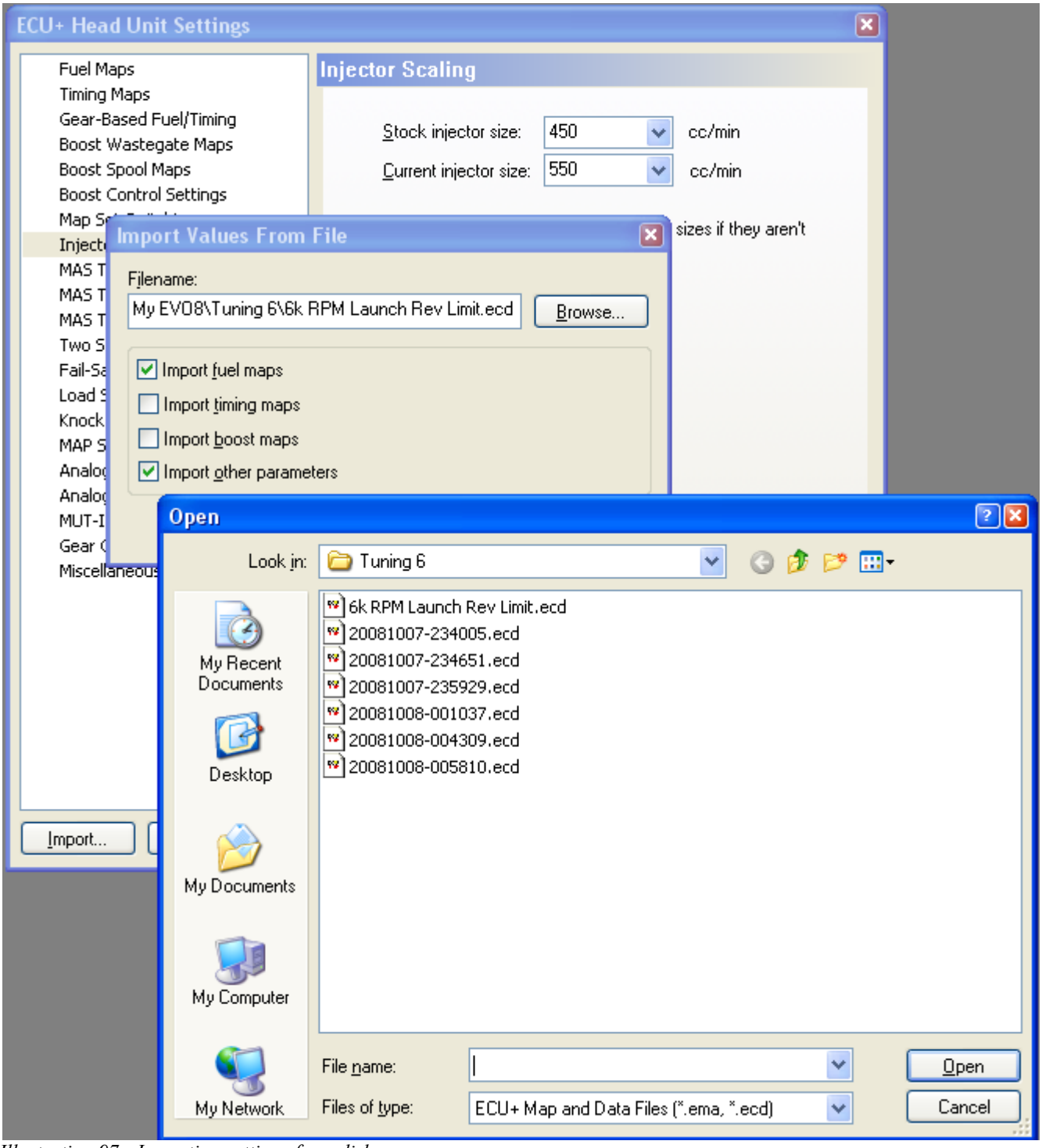


Illustration 97 - Importing settings from disk

By default, saved head unit settings files have the .ema file extension, though you can also load the values used in a previous capture by specifying a .ecd capture file here. The ECU+ Win software remembers the settings used in a previous capture, and you can import settings from that type of file.

7.12 Changing Head Unit Settings Offline

In general, editing the ECU+ head unit settings requires a laptop and a serial connection to a powered-up ECU+ head unit. However, some people prefer to do analysis and modification of settings while

“offline” - disconnected from the ECU+ head unit. The ECU+ Win software enables this mode of operation as well. To use this feature, select the *Settings->ECU+ Head Unit (Disconnected)* menu item.

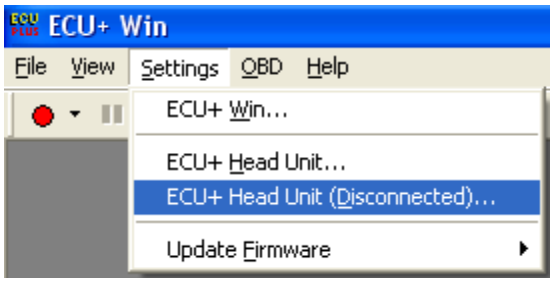


Illustration 98 - The head unit settings disconnected menu item

This brings up an offline editor for the ECU+ head unit settings. The offline editor works exactly the same as the normal head unit settings editor, except that the settings can only be imported or saved to disk. Later, when the ECU+ is connected, you can import these offline-edited settings into the head unit.

7.13 On Board Diagnostics (OBD)

The ECU+ includes a complete diagnostics module that works with the stock ECU on all DSMs and the Evolutions. The diagnostics module allows the ECU+ to communicate in real-time with your stock ECU, and display Diagnostic Trouble Codes (DTCs, which indicate the failure of various vehicle systems), as well as real-time and freeze frame ECU data, like RPM and throttle position. This diagnostic information is normally accessible only via expensive dedicated “scan tools,” but the ECU+ now includes a module that can access and display the same information that these expensive scan tools can, and the ECU+ also features real-time display of engine parameters and trouble codes.

The ECU+ supports the Mitsubishi-specific MUT-II protocol as well as the SAE (Society of Automotive Engineers) OBD-II protocol. MUT-II works on any Mitsubishi ECU, while OBD-II is only supported in 1996 or newer vehicles. MUT-II displays information more quickly than OBD-II, but doesn't include the extended diagnostic information that OBD-II does. Generally, you'll want to use MUT-II on 1st generation DSMs, and OBD-II on later vehicles.

To start the ECU+ Win OBD displays, use the *OBD* menu item. The *OBD* menu item will display information in either MUT-II mode or OBD-II mode depending on what mode you've configured in the OBD mode setting in the ECU+ Win settings dialog. More information is available in the section Miscellaneous Configuration on page 53.

The available displays are as follows:

7.13.1 Information

This display is a real-time engine data monitor, and displays all of the information that the stock ECU exports via OBD-II or MUT-II. This information is split into three sub-displays: Basic, Advanced (OBD-II only) and Tests (OBD-II only).

7.13.1.1 Basic (OBD-II and MUT-II Modes)

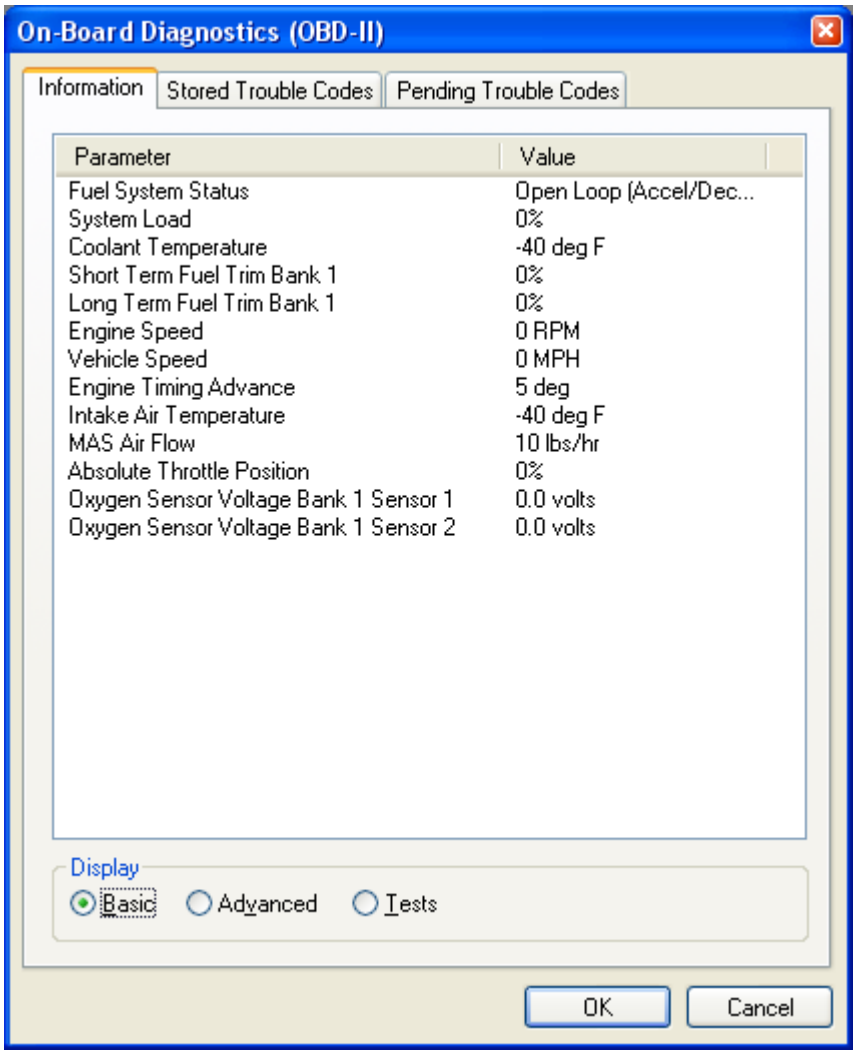


Illustration 99 - The basic information tab (OBD-II mode)

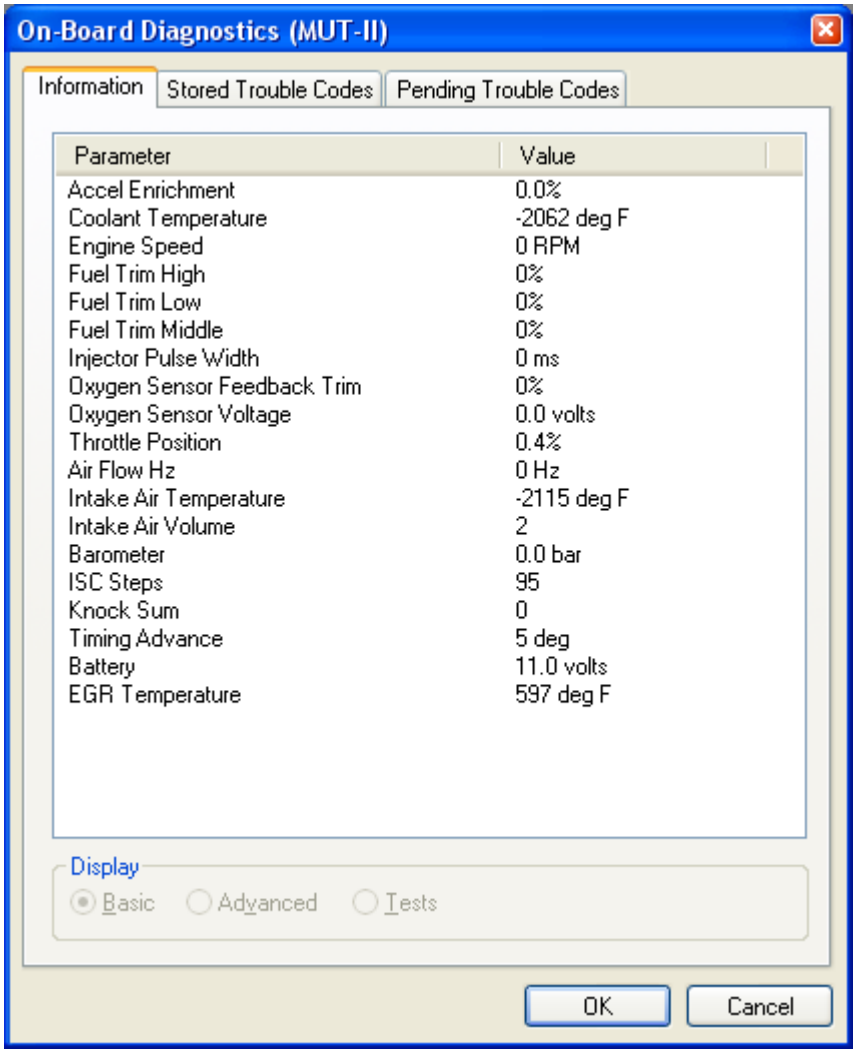


Illustration 100 - The basic information tab (MUT-II mode)

The basic information tab is a real-time display of various vehicle parameters *as seen by the stock ECU*. Much of the information displayed here duplicates data that is available from the ECU+, but it can still come in handy when debugging problems with your vehicle. An example might be that your cruise control isn't working, and you suspect (or have a trouble code to indicate) that your speed sensor is bad. By looking here, you may find that the stock ECU doesn't ever see a vehicle speed over zero, and yet the ECU+ registers vehicle speed just fine. That may indicate that the speed sensor wire is broken between the ECU+ tap point and the stock ECU.

Other values displayed here are in addition to what the ECU+ can display or datalog. Some of these are:

- *Fuel System Status* (OBD-II mode only) – the stock ECU can run in one of two modes when controlling fuel to the engine: open loop or closed loop. In open loop mode (used at WOT or high acceleration conditions), the stock ECU looks up a fuel flow number in a table and turns on the injectors for a fixed amount of time based on air flow, throttle position and the values of the temperature and pressure sensors. In closed loop mode, the stock ECU hunts for a perfect air-fuel ratio by monitoring the front O2 sensor and adjusting fuel flow dynamically. When you see your front O2 sensor voltage cycling up (to around 1 volt) and down (to near zero volts), that's the stock ECU hunting for a perfect (14.7:1) air-fuel ratio by alternately running slightly rich (front O2 voltage around 1 volt) and then slightly lean (front O2 voltage near zero volts). With this display, you can be sure what mode the stock ECU is using. The display will show one of four values:

  - Open Loop (Normal) – either the front O2 sensor or the engine hasn't reached operating temperature yet.
  - Open Loop (Accel/Decel) – vehicle is accelerating or decelerating.
  - Closed Loop (Normal) – normal operation for low load driving.
  - Open/Closed Loop (System Fault) – a sensor problem has been detected.
- *System Load* (OBD-II mode only) – this is a percentage of how hard the engine is running. 0-100%.
- *MAS Air Flow* (OBD-II mode only) – MAS air flow is a computed number inside the stock ECU that takes into account the MAS frequency (what you see on the ECU+'s display), intake air temperature and barometric pressure. This number is displayed as a flow rate into the engine, in lbs/hr.
- *Intake Air Volume* (MUT-II mode only) – this is the MUT-II equivalent to OBD-II's *MAS Air Flow*.
- *Absolute Throttle Position* (OBD-II mode) or *Throttle Position* (MUT-II mode) – this is the same information as the ECU+'s throttle position sensor voltage, except that it's displayed as a percentage rather than a voltage. The stock ECU has fixed low and high limits for the voltage corresponding to "0%" and "100%" open throttle, and displays an interpolated value here based on the true throttle position sensor voltage.
- *Oxygen Sensor Voltage* – bank 1, sensors 1 and 2 (in OBD-II mode) should correspond exactly to the front and rear O2 sensor voltages as seen by the ECU+. The OBD-II specification allows a vehicle to contain more than two O2 sensors, and so that specification categorizes the O2 sensors according to what engine bank and exhaust position the sensors are placed at. The ECU+'s OBD-II module follows that naming convention rather than displaying the values as front/rear O2 sensor voltages. In MUT-II mode, this is the front O2 sensor voltage.
- *Short and Long Term Fuel Trims* (OBD-II mode) and *Low, Medium and High Fuel Trims* (MUT-II mode) – these are the most important parameters displayed here. As outlined above, the stock ECU can run in either open or closed loop mode. When in closed loop mode, the stock ECU will adjust its fuel flow based upon feedback from the front O2 sensor. The stock ECU contains a baseline set of fuel tables, pre-programmed at the factory for the stock injectors. It uses these fuel tables as a starting point in calculating the optimum fuel flow for a given air flow (and temperature/pressure readings) in closed loop mode. From that starting point, the stock ECU will continuously (over a few seconds worth of time) add, then subtract fuel in an effort to maintain a perfect (14.7:1) average air-fuel ratio. But the stock ECU needs a mechanism to adapt for injectors that are out-of-spec or clogged, and that is the short term fuel trim. You can think of the short term fuel trim (OBD-II mode, low trim in MUT-II mode) as an offset from the pre-programmed stock ECU fuel table values whose job it is to compensate for imperfect injector flow rates.

  With the ECU+, you're modifying the MAS air flow signal to influence fuel flow from the stock ECU. At WOT (in open loop mode), where most tuning is done, the fuel trims don't come into play. Under low and medium loads, though, the stock ECU uses these fuel trims to maintain a perfect air-fuel ratio and thus give you good driveability and gas mileage. The fuel trims have a limited range, and if your car isn't tuned properly under closed loop conditions, the fuel trims will max-out in one direction or the other, thus causing your engine to run consistently lean or rich in closed loop mode.

  By monitoring the fuel trim display, you can see how well your engine is tuned in closed loop mode. In general, you'll want to tune the ECU+'s fuel map for idle and low and medium loads such that the fuel trims are near 0% when the car is operating in a given range. If you find that the fuel trim is a negative value for a given range, then your engine is running rich in that range and you should subtract fuel in that range of the ECU+'s fuel maps. Conversely, a positive short term fuel trim number indicates that the engine is running lean in that range, and you should add fuel with the ECU+'s fuel maps.

In general, you'll want to watch the short term (low in MUT-II mode) fuel trim. The long term fuel trim (high in MUT-II mode) is, as its name implies, a longer-term average of the short term fuel trim. As with the short term trim, this value should stay around 0%. By occasionally checking the long term fuel trim, you can be alert to changes in your injector performance that may require re-tuning your car.

You may find fewer, or more, items displayed for your car than is shown in the screen shots. As with all of the OBD-II screens, the ECU+ queries the stock ECU to see what values are supported and displays only the supported data.

7.13.1.2 Advanced (OBD-II Mode Only)

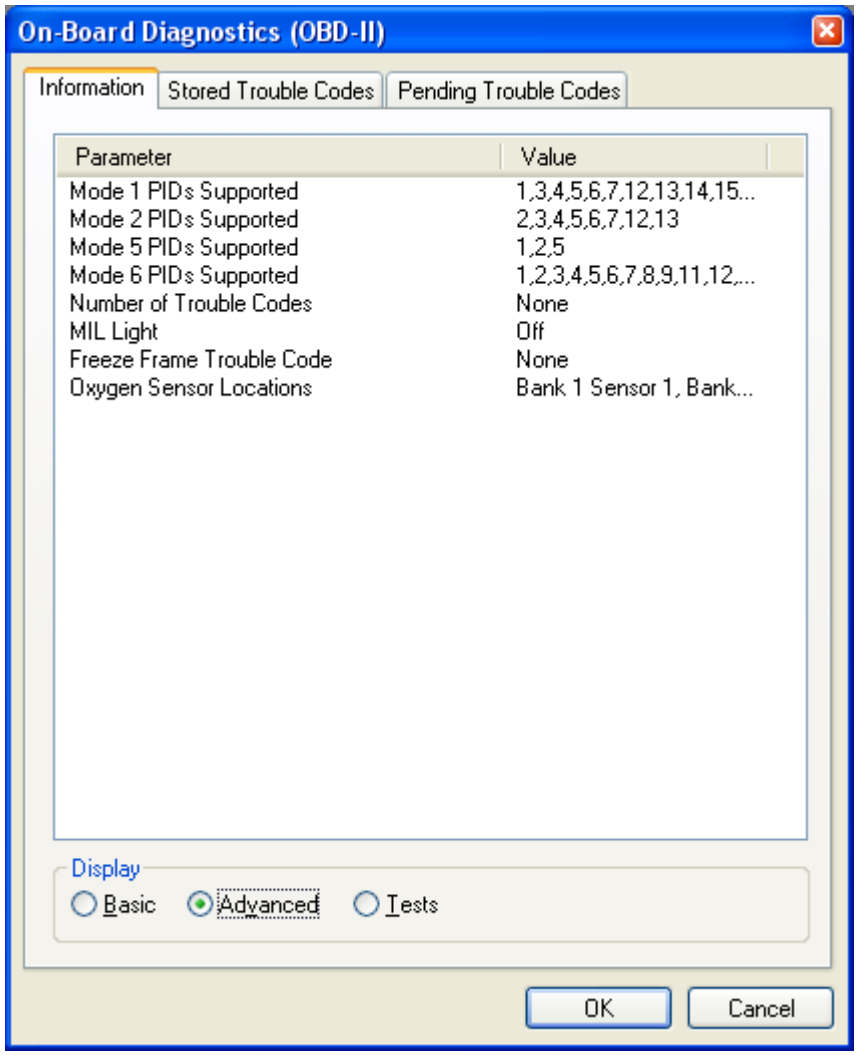


Illustration 101 - The advanced information tab

The advanced information tab is another real-time display which shows advanced OBD-II information that may be useful for debugging software problems with the ECU+'s OBD-II implementation. Here's some of the things you'll see:

- *Mode N PIDs Supported* – this is a cryptic list of "PIDs" (parameter IDs) available in the various OBD-II modes that your stock ECU includes. Different OBD-II modes perform different functions, and the ECU+ needs to know what sub-functions are supported for each mode. The ECU+ Win software displays this as a list of numbers. The most interesting of these is the mode 1 PIDs, which specify which values will display on the basic information screens.

- *Number of Trouble Codes* – this is the number of Diagnostic Trouble Codes (DTCs) currently stored in the stock ECU.
- *MIL Light* – whether or not your MIL (malfunction indicator light, or check engine light) is on due to a stored trouble code.
- *Oxygen Sensor Locations* – which oxygen sensor are supported in this vehicle.
- *Design OBD Requirements* – which OBD specification this stock ECU supports. Potential values include the early-model OBD, OBD-II as well as the European and Japanese OBD specs.

7.13.1.3 Tests (OBD-II Mode Only)

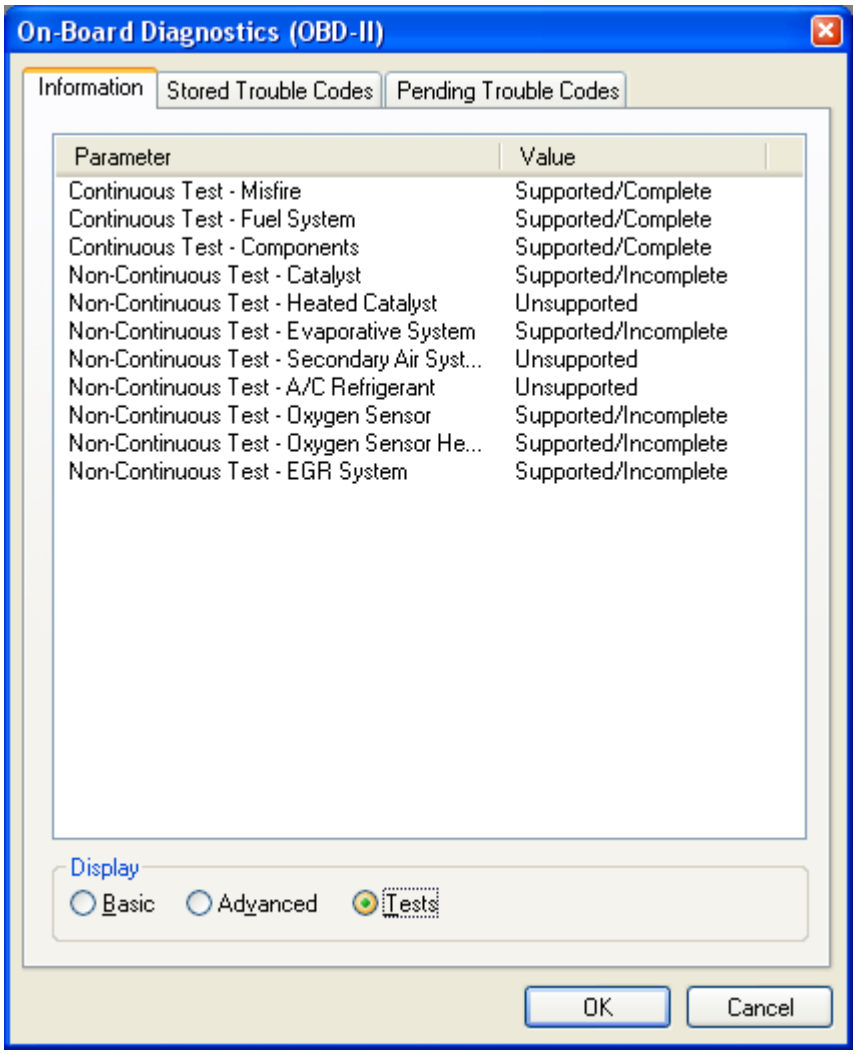


Illustration 102 - The tests tab

The OBD-II specification requires manufacturers of vehicles to support various internal tests for (generally) emissions-related functions. These tests determine whether the vehicle is operating properly from an emissions point of view. The tests information tab displays the results of these tests. You'll find two types of tests: continuous and non-continuous. Continuous tests run continuously whenever the engine is running, while non-continuous tests must be invoked manually with a scan tool. For each of the tests, the ECU+ will display whether or not the test is supported by the stock ECU, and if the test is supported, whether it has run to completion.

7.13.2 Stored Trouble Codes and Freeze Frame Data (OBD-II and MUT-II Modes)

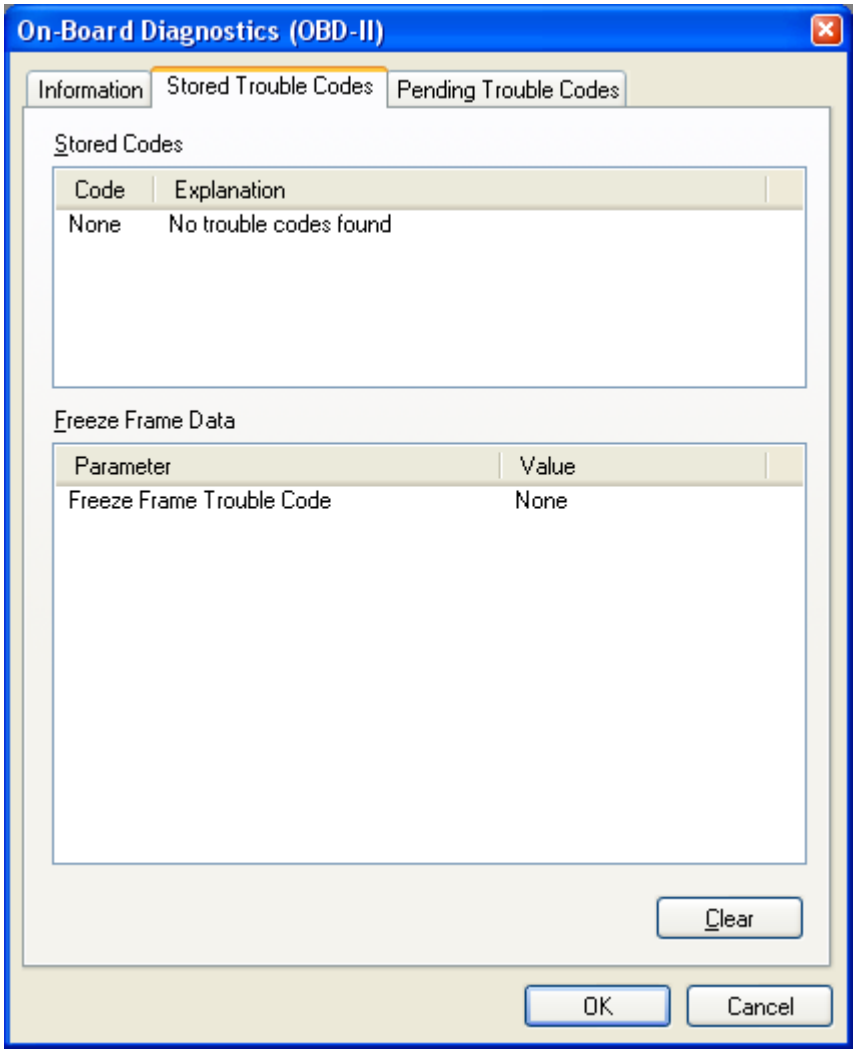


Illustration 103 - The stored trouble codes tab

Diagnostic Trouble Codes (DTCs) are how the stock ECU flags problems with your engine. When a problem is detected by the stock ECU, it sets a trouble code and turns on the MIL (malfunction, or check engine light) to indicate that a problem has occurred. Problems can be simple, like no signal from a particular sensor, or complex, like "random misfire detected." These trouble codes can be displayed and cleared by the ECU+.

Two types of trouble codes exist inside the stock ECU: pending trouble codes (OBD-II mode only) and stored trouble codes. Pending codes indicate that a problem was detected, but that the problem is not (yet) considered to be important enough to turn on the check-engine light. Think of the pending codes as an early-warning that something may be wrong. When a pending code occurs frequently, or on multiple drive cycles, the pending code will be "promoted" and become a stored trouble code. Pending codes that only occur infrequently are typically forgotten after one or two additional drive cycles.

On the stored trouble codes tab, the ECU+ displays the stored trouble codes currently in effect. All of the codes are displayed, along with the standard trouble code description (note that MUT-II mode only displays a small subset of the potential trouble codes). Additionally, a set of freeze frame data may also be displayed (OBD-II mode only). Freeze frame data is stored when an emissions-related code is stored. This freeze frame data is a snapshot of what the engine was doing at the time the code was set,

and is very useful in tracking down the source of the problem. If multiple stored codes are present, the freeze frame data applies (generally) to the first code that was seen, and the "freeze frame trouble code" indicates which stored code is associated with the freeze frame data.

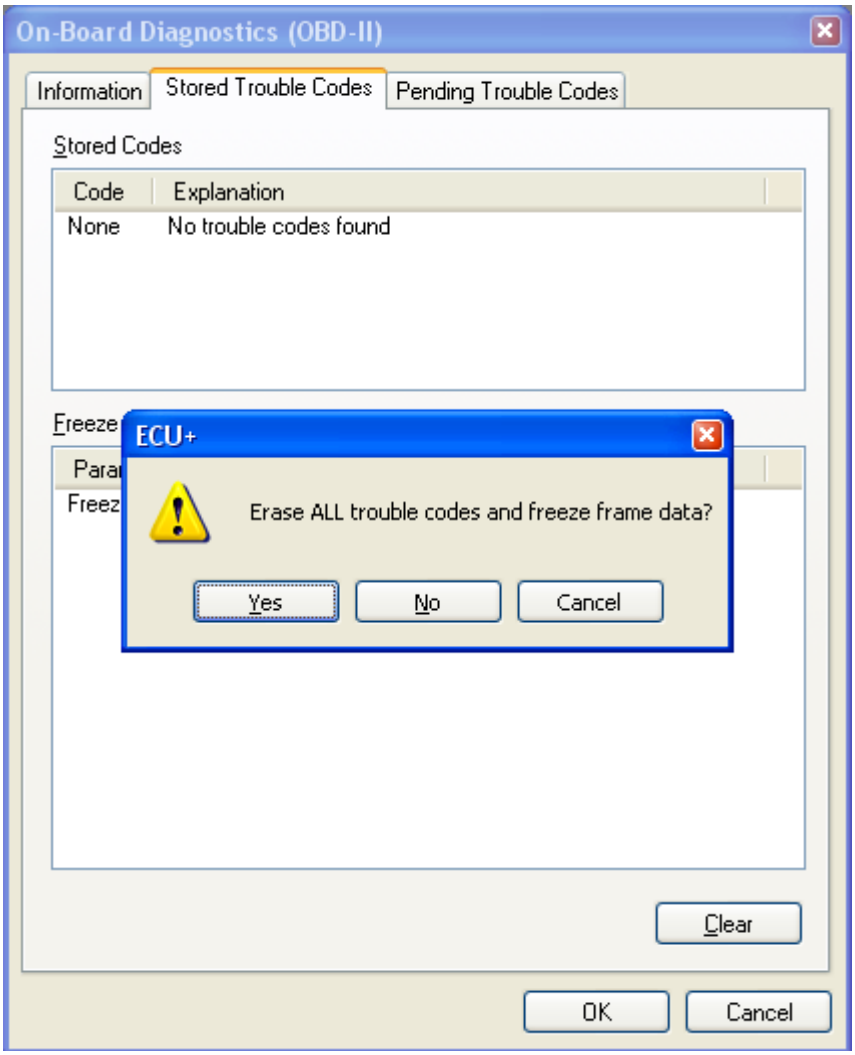


Illustration 104 - The code-clearing confirmation prompt

On this tab, a button is provided that allows the trouble codes to be cleared. When invoked, this will erase all stored and pending trouble codes as well as the freeze frame data. It will also extinguish your MIL light.

7.13.3 Pending Trouble Codes (OBD-II Mode Only)

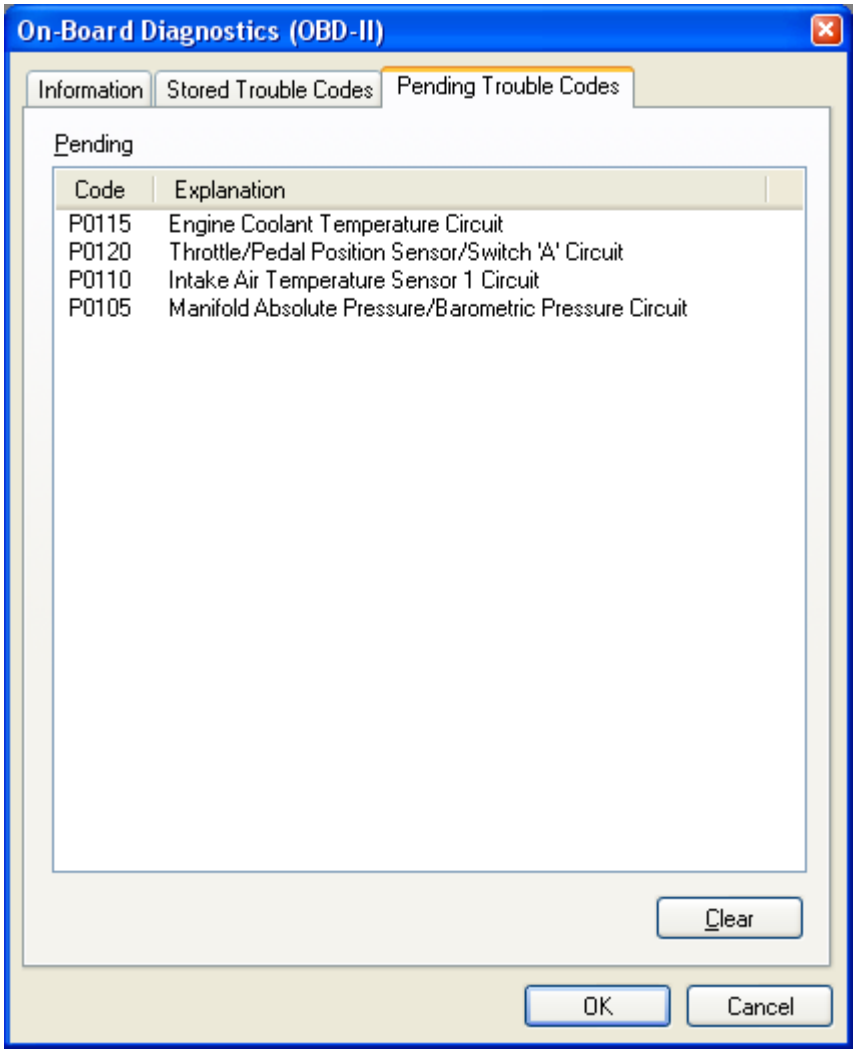


Illustration 105 - The pending codes tab

The pending trouble codes tab shows all of the codes that are currently pending, before they are promoted to stored status. As with the stored trouble codes tab, a clear button is provided to allow these codes to be cleared.

7.13.4 Using the ECU+ With Other Code Scanners and the EcuFlash Tool

To implement the OBD-II and MUT-II functionality, the ECU+ connects directly to the OBD protocol pins at the stock ECU. These pins also connect to an under-dash (near the steering wheel) connector so that external OBD-II or MUT-II scan tools (the latter includes the EcuFlash software/hardware combination) can talk to the stock ECU. Normally, the OBD port is a single master interface (the stock ECU is the slave), and the ECU+ is that master. This can cause a problem when another device wants to talk to the stock ECU on the OBD pins, as the ECU+ expects to be the master device. As designed, the ECU+ firmware will do its best to avoid this situation, as the firmware software-disconnects from the port when not doing datalogging or displaying an OBD-II/MUT-II screen. However, some external devices are very sensitive and may not work when the ECU+ is physically connected to the OBD pins. This issue can rear its head in several common situations:

- If you take your car to the dealer or other repair facility, their scan tool may not work as long as

the ECU+ is installed.

- Some states are using the OBD-II port to initiate tests for emissions compliance. With the ECU+ installed, these tests may not run and your vehicle will fail emissions testing.
- The EcuFlash software and hardware may not be able to read and burn maps into your stock ECU with the ECU+ installed.

There are two ways to handle these problems:

1. In the ECU+ wiring harness, locate pin 22 (black wire, group C) on the low current harness and snip the wire in half. Then insert a quick-disconnect in this wire that will allow you to easily disconnect this wire whenever you need to access the OBD-II functionality with an external scan tool.
2. Inside of the ECU+ head unit, there is an internal jumper that will serve this same purpose. Remove four screws from each end of the ECU+ head unit, and slide out the printed circuit board. Near the high current (white) connector is a jumper, JP8. Simply remove the shorting block from this jumper and reassemble your head unit.

7.14 ROM Editing and Map Tracing

The ECU+ includes an advanced software module that allows the ECU+ to work cooperatively with the EcuFlash software for the Evolution VIII and IX. In this mode of operation, the EcuFlash software is used to download and burn "ROM" files from your stock ECU, and the ECU+ Win software is used to edit the ROM file within the normal ECU+ environment. (ROM files are binary copies of the software inside your stock ECU. These include many "maps" of calibration constants that the ECU+ can modify.) Additionally, the ECU+ can use datalogged information to trace out the current value of the stock ECU maps so it's easy to see what the best values to edit within the ROM files are.

7.14.1 Downloading Your Stock ECU ROM

The first step in using the ECU+'s ROM editing and map tracing is to download the contents of your stock ECU using the EcuFlash tool. Save the contents as a .bin file. Refer to the EcuFlash documentation for the steps involved in downloading this file.

The contents of your stock ECU is saved to a so-called "ROM file" that you'll be editing and tracing inside the ECU+ software. Note that in rare cases the EcuFlash tool may not be able to retrieve your stock ECU contents when the ECU+ is connected. See the section titled Using the ECU+ With Other Code Scanners and the EcuFlash Tool on page 116 for techniques available to correct this problem.

7.14.2 Configuring ROM Editing

To configure ROM editing, you'll need to tell the ECU+ software two pieces of information:

1. The name and location of your stock ECU ROM file.
2. Where the EcuFlash *rommetadata* directory (for your car type) is on your hard disk. This directory tells the ECU+ software the location of various XML files that will be parsed to determine where the "maps" are within your stock ECU ROM file. The directory should contain a number of .xml files.

To enter this information, use the ECU+ menu item *Settings->ECU+ Win* and select the *ECU ROMs* dialog as shown in Illustration 106 below.

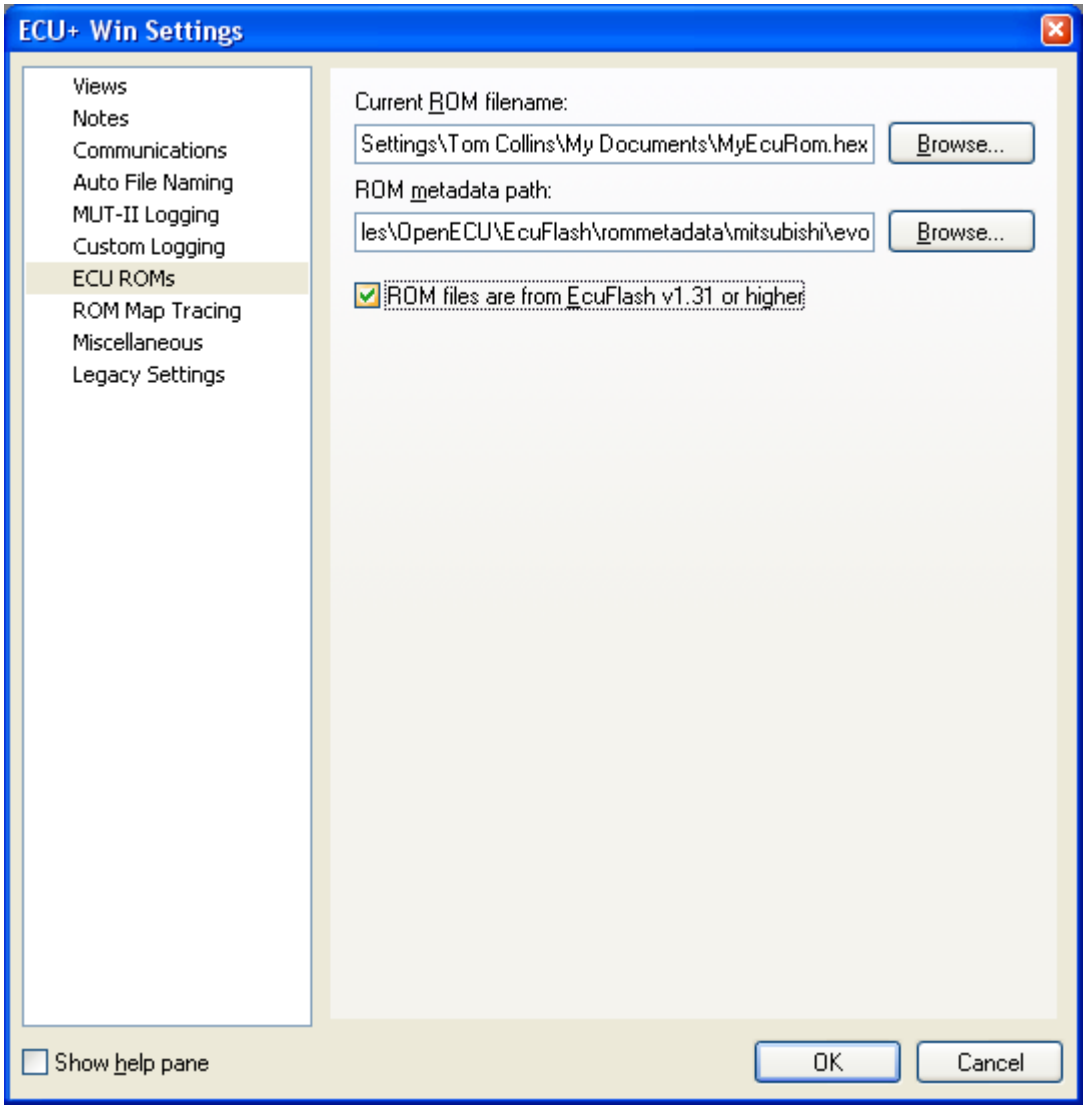


Illustration 106 - The ECU ROMs configuration dialog

Additional information about this dialog is in the section ECU ROMs Configuration on page 50.

7.14.3 Using ROM Editing

To actually use the ECU+'s ROM editing capability, simply open an existing capture file on disk. After opening the capture file, ECU+ Win will load your stock ECU ROM file into memory and look through the metadata XML files to find one that describes your ROM file. Potential problems and their solutions are as follows:

1. ECU+ Win won't be able to load your ROM file and will issue a *Error opening ROM file <filename>* error. This usually means that the *Current ROM filename* specified above is not there.
2. ECU+ Win may not be able to parse one of the metadata XML files. This usually indicates that the XML was hand-edited (incorrectly) and is not a valid XML file. In general, the XML files included in a EcuFlash distribution should always parse correctly, but files that you get from other sources may be less well-formed. ECU+ Win will display a meaningful error message (meaningful, at least, to someone who understands the XML format of these files) in this case. Note that after confirming the error, ECU+ Win will continue to parse the other XML files in

the named directory, so this isn't a fatal error, but you'll want to get to the bottom of the error nonetheless. “Fixing” the problem usually involves correcting bad XML tags or just deleting the problem file.

3. ECU+ Win may not be able to find a matching XML description of your stock ECU ROM. In this case, the software will issue a *Couldn't locate a matching ROM metadata file for your ROM* error. One of the metadata XML files is supposed to include an ID value that matches your stock ECU ROM file. In this case, ECU+ Win looked through all of the XML files and couldn't find a match. The only solution to this problem is to find someone who has a good definition of your ROM and load that XML file into the EcuFlash metadata directory. Note that you'll also see this error if you incorrectly specify the *ROM metadata path* in the above dialog. This path should generally end in “evo” for the Evolution VIII and IX's.

If there are no errors, ECU+ Win will simply display your capture file as always. To actually use the ROM editing feature, right-click on a view window and select *ECU Maps* from the context menu as shown in Illustration 107.

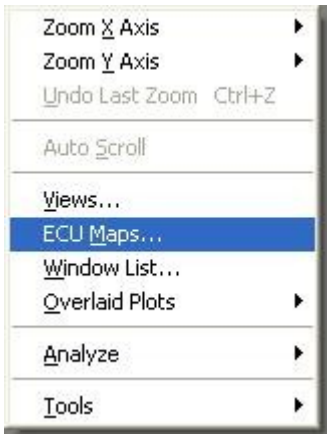


Illustration 107 - The ECU Maps context menu item

This will bring up the ECU Maps dialog box as shown in Illustration 109.

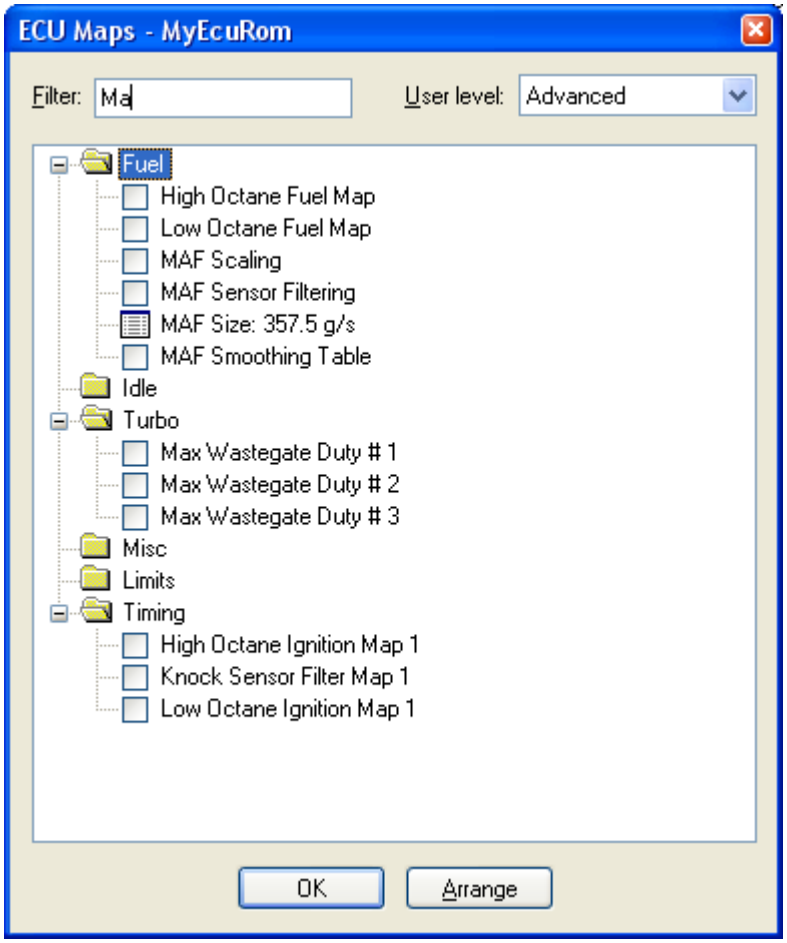


Illustration 108 - The ECU Maps dialog box

This dialog box lists all of the "maps" available within your stock ECU ROM. In general, these correspond to lookup tables of parameters that the stock ECU uses when running the engine. The maps are listed in a tree configuration, with related maps grouped together under a common folder. Two features are included here to make locating maps easy:

- A *User level* dropdown, which causes the dialog box to only show a subset of the available maps.
- A *Filter* text field, which causes the dialog to only show the maps that include the string that you type into the *Filter* field. Illustration 109 shows this feature in action. Note that when using a filter, the folders that include matching maps will spring open to display all of the matching maps.

ECU+ Win will remember your *User level* and *Filter* settings from one session to another.

Within the dialog box, some maps (like MAF Size, above) will have a grid icon next to them. These are so-called "1D" maps, where the "map" is just a single editable value. Double-click the value to edit the value, and press ENTER when done.

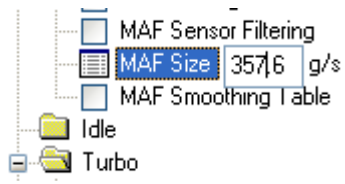


Illustration 109 - Editing a 1D map

Most of the maps, however, are so-called “2D” or “3D” maps. These have a checkbox next to them. Checking the box will pop up the map in the display area of ECU+ Win. Un-checking the checkbox will remove the map from the display area. Use the *Arrange* button to cascade the displayed maps, and the *Ok* button to close the ECU Maps dialog box.

To edit a 2D (one column, or rarely, one row of values) or 3D (a rectangular grid of values) map cell, simply double-click inside of the cell with the mouse, change the value, and press ENTER. Changed cells are displayed in a bold font. Note that when editing values, the ECU+ Win software may modify the value that you enter to conform to the format and precision of that cell. To edit more than one cell, select multiple cells with the mouse or keyboard. Shortcuts for selecting multiple cells are identical to those used when editing the ECU+ fuel and timing maps, and are as follows:

Action	Mouse	Keyboard
Select a row of cells	Click an row header cell	
Select a column of cells	Click a column header cell	
Select all cells	Click the *cell in the top left*	
Select a rectangular block of cells	Click the first cell, then shift+click the last cell	With the cursor in the first cell, use shift+up/down/left/right arrows
Select multiple disconnected cells	Click the first cell, then ctrl+click another cell	

Table 7- Available shortcuts to select multiple cells

With multiple cells selected, right-click on a map to bring up the map editing context menu.

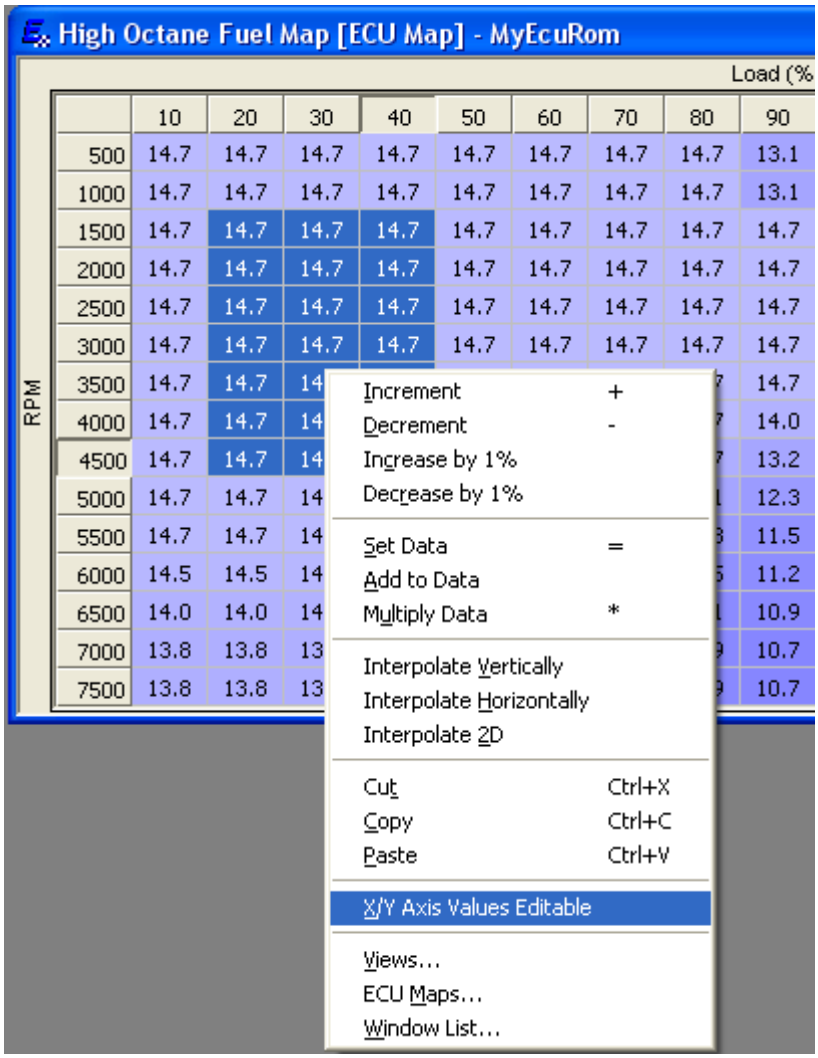


Illustration 110 - The map editing context menu

Most of the entries on this menu are identical to those on the fuel and timing map editing context menu, and are self-explanatory. Note that *Increment* and *Decrement* alter cells by one least-significant digit (14.7 to 14.8 for example), while the *Increase/Decrease by 1%* and *Multiply Data* menu items can scale the cells by any fraction that you'd like.

The *X/Y Axis Values Editable* menu item is unique to the ECU maps editing capability. It turns out that the X and Y axis values within an ECU map are, in fact, other ECU maps. As such, the axis values can often be edited. For example, in Illustration 110, it's possible to change the 5500 RPM value to 5600 RPM so that the 5500 RPM row of values will activate at 5600 RPM. To enable this editing, simply select the *X/Y Axis Values Editable* menu item and the axis values will change so that they are editable.

If you make changes to any of the ECU maps, when you close your capture file you'll be prompted as to whether to save your changes to the ECU maps. If you say yes, any changes you have made to your stock ECU ROM file will be written to disk.

7.14.4 Uploading Your Modified Stock ECU ROM

After you've used the ECU+'s ROM editing feature to alter your stock ECU ROM file, you'll need to upload your ROM file to your stock ECU using the EcuFlash tool. Refer to the EcuFlash documentation for the steps involved in uploading this file. A future release of the ECU+ software may natively include this ROM uploading and downloading capability, but for now you'll need to use the

EcuFlash software to do these functions.

7.14.5 Configuring Map Tracing

Map tracing is a software function that allows the ECU+ Win software to highlight, or “trace” the ECU map cell that's being used by the stock ECU at a given point within an ECU+ capture file. ECU+ Win inherently knows how to trace its internal fuel and timing maps, so there's no configuration involved there. But with the stock ECU ROM files, the ECU+ Win software doesn't inherently understand what the X and Y axis' on a stock ECU ROM are (these are configurable via the XML metadata files), and thus doesn't know what cells to highlight for a given set of engine conditions. When you configure map tracing, you're telling the ECU+ Win software what the X and Y axis' (within a given map) correspond to, in terms that the ECU+ Win software knows about (like RPM or throttle position).

To configure map tracing, use the ECU+ menu item *Settings->ECU+ Win* and select the *ROM Map Tracing* dialog as shown in Illustration 111 below.

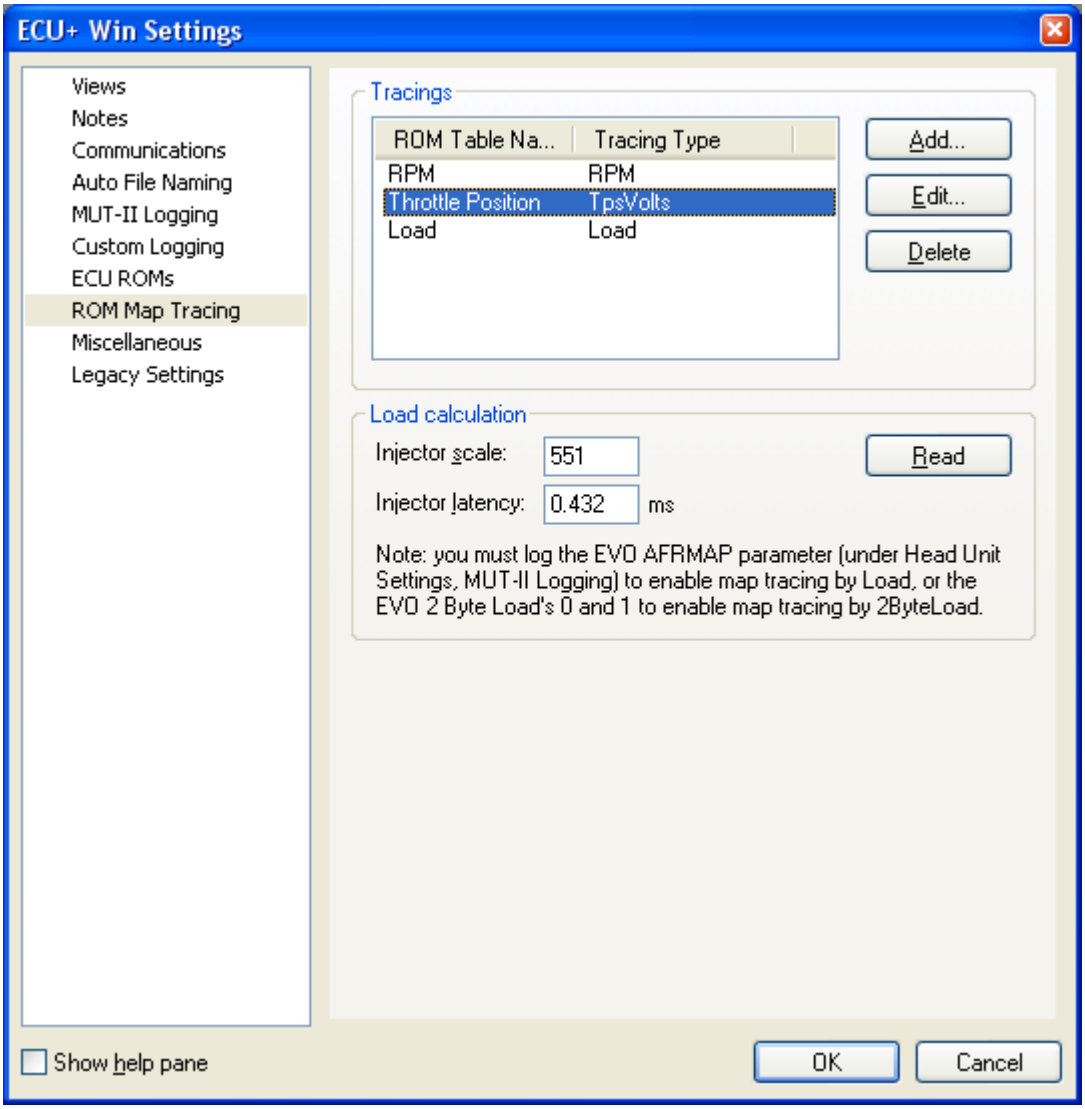


Illustration 111 - The ROM map tracing configuration dialog

Inside the *Tracings* section of this dialog, you'll add entries that correlate a ECU ROM map X (or Y) axis table with an internal ECU+ variable. To do so, click the *Add* button to bring up the *Configure Tracing* dialog as shown in Illustration 112.

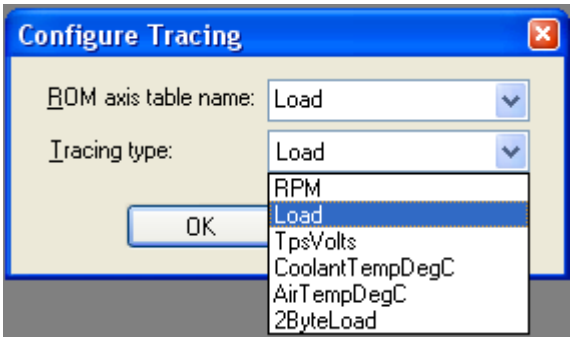


Illustration 112 - The configure tracing dialog

Select a *ROM axis table name* from the dropdown (this is name of an X or Y axis from the internal XML metadata file corresponding to your configured ECU ROM file), and then a *Tracing type* from the other dropdown and click *Ok*. The *Tracing types* that the ECU+ software currently supports are:

- *RPM* – the current engine RPM.
- *Load* – a calculated "engine load" value. Load is calculated using the EVO AFRMAP MUT-II parameter, which must be logged to enable tracing by Load.
- *2ByteLoad* – a higher resolution "engine load" value. Load is calculated using the EVO 2 Byte Load 0 and EVO 2 Byte Load 1 MUT-II parameters, which must be logged to enable tracing by 2ByteLoad.
- *TpsVolts* – the throttle position sensor voltage.
- *CoolantTempDegC* – the engine coolant temperature, in degrees C.
- *AirTempDegC* – the intake air temperature, in degrees C.

The *Load* tracing type is special. Load is computed inside of the ECU+ Win software, but the calculation needs specific values for the injector scale and injector latency values used inside the stock ECU. Enter new values in the text field, or use the *Read* button to read the values from current ROM file.

Some things to note about configuring map tracing:

1. The *ROM axis table name* dropdown is populated with the unique X and Y axis names corresponding to the ECU ROM file that you've configured. If you haven't configured a ROM filename, or the file is unreadable, this dropdown will be blank.
2. You must save the ECU+ Win configuration settings once (by pressing *Ok*) after changing the ROM filename before the *ROM axis table name* dropdown is populated correctly.
3. Ditto for using the *Read* button to pull the injector scale and injector latency from the configured ECU ROM file.
4. More tracing types can be easily added in the future. E-mail suggestions to support@ecuplus.com.

7.14.6 Using Map Tracing

With map tracing configured, using map tracing is simple: just bring up the ECU Maps dialog and select the maps you'd like to see. If map tracing is configured correctly, the ECU+ Win software will immediately highlight the associated ECU map cell as the cursor is moved inside of a capture file. See Illustration 113 for an example.

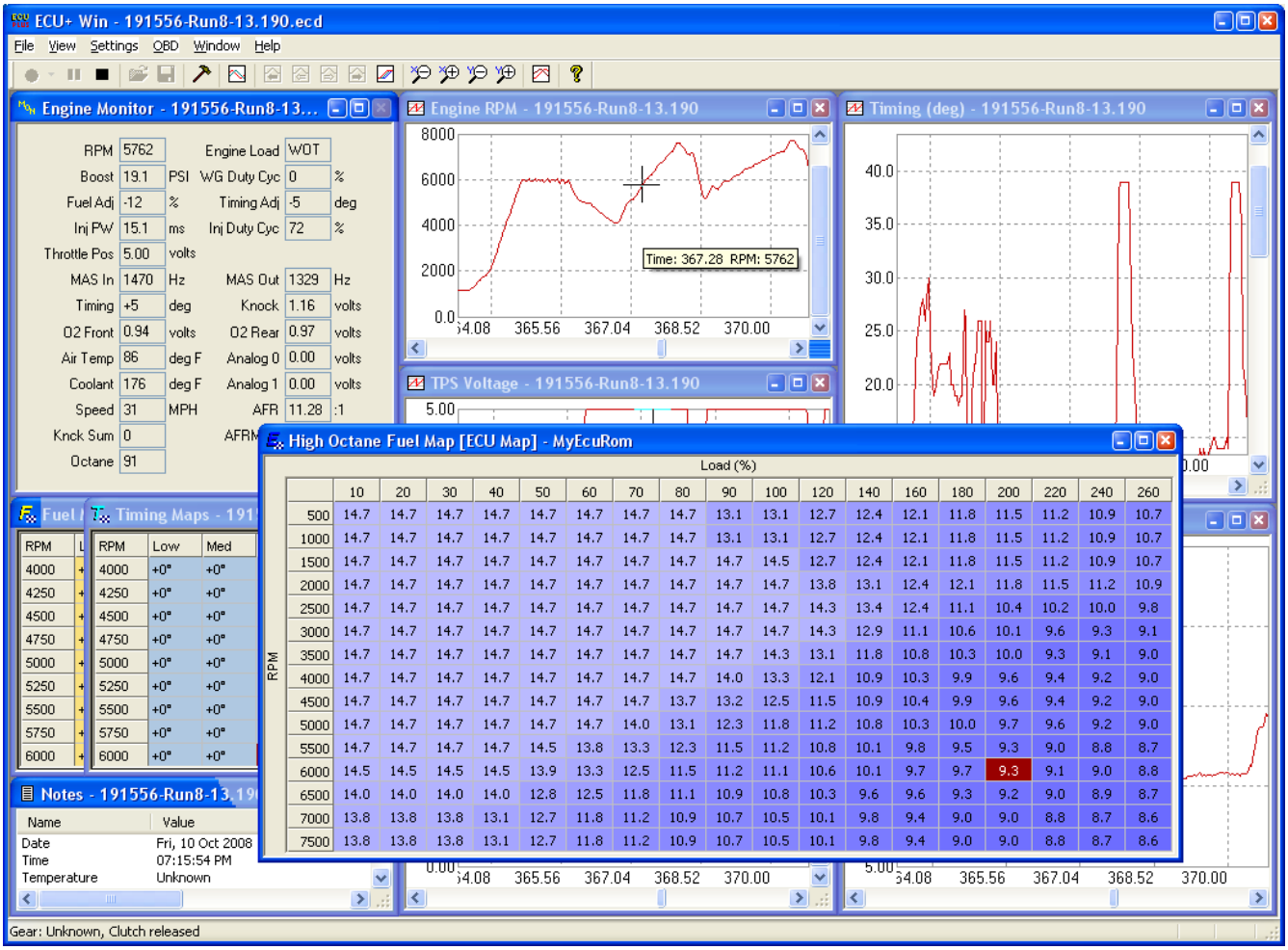


Illustration 113 - ROM map tracing in action

8 Software Utilities

The ECU+ software suite includes a command-line utility for debugging serial port problems. This utility is described in this section.

Note to International customers: the ECU+ software installs by default to the directory %ProgramFiles %\EcuPlus. This is C:\Program Files\EcuPlus in the English version of Microsoft Windows, but may be different in other languages. All of the directory names listed below assume an English install of Windows, where the ECU+ software is installed in C:\Program Files\EcuPlus.

8.1 The Serial Port Tester, SerTestW

To diagnose problems when connecting and performing captures, the ECU+ software suite includes a utility called *SerTestW*. *SerTestW* is a simple utility that iterates through all of the serial ports installed on your laptop, checks to see if a ECU+ head unit is connected, and if so, exhaustively tests whether the serial port can read and write data to the head unit. Results are displayed on-screen and logged to a log file should you need to send these results in to support@ecuplus.com.

To run *SerTestW*, follow these steps:

1. Using the Windows Start menu, navigate to *Start->Run*, and then execute the command *cmd* as shown below.

Illustration 114 - The Windows Start menu

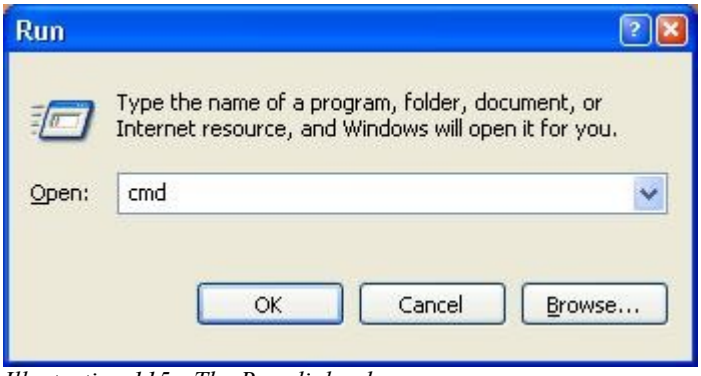


Illustration 115 - The Run dialog box

2. Change to the directory *C:\Program Files\EcuPlus\Win32* and start the program by typing *SerTestW* and then ENTER.

The software will run to completion and logs its results to a log file named *sertest.log*. A screenshot of a sample run of *SerTestW* is shown below. This run shows that *SerTestW* found a ECU+ head unit on COM1, and communications were successful.

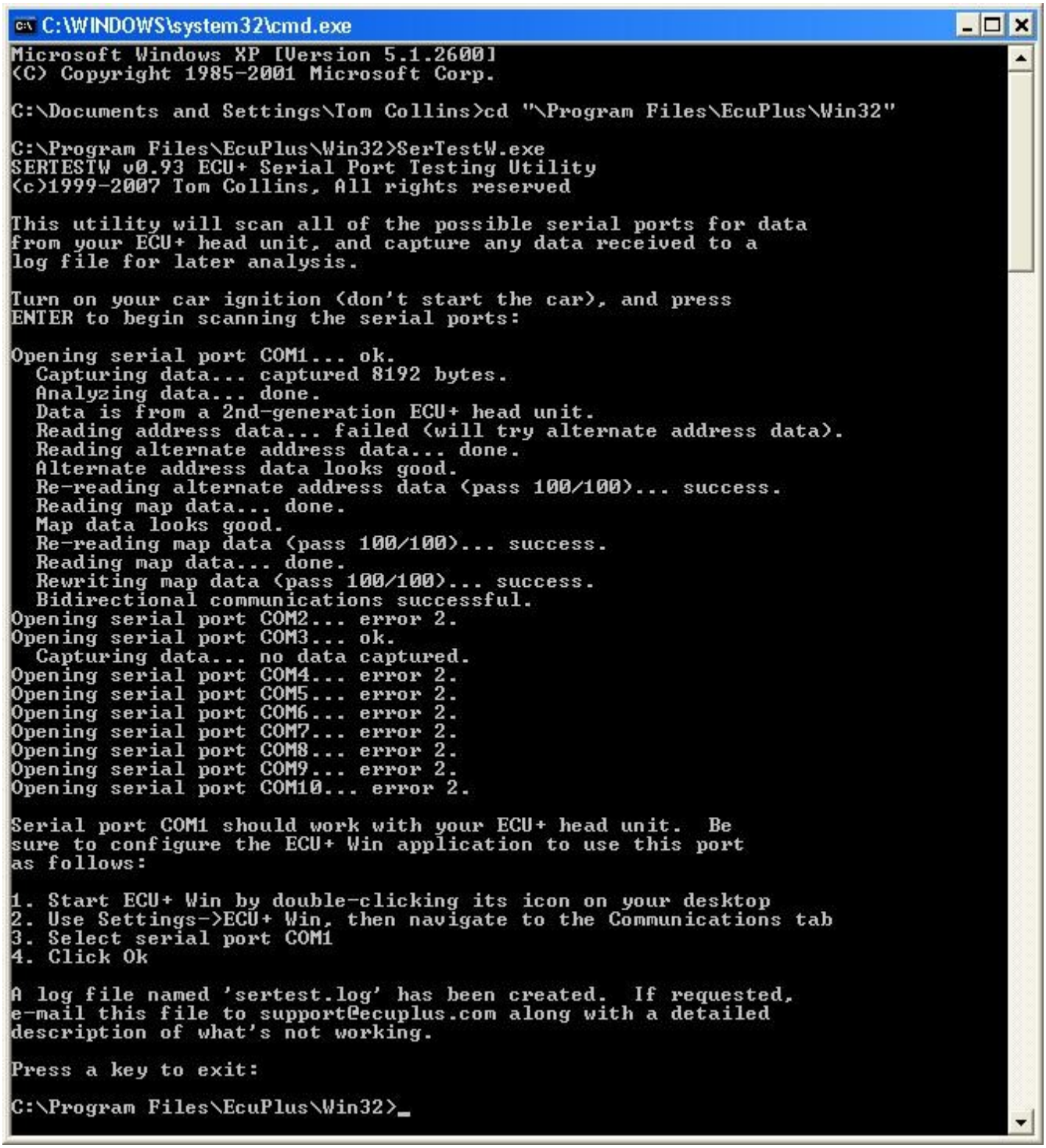


Illustration 116 - A sample run of SerTestW

9 Upgrading Your Head Unit Firmware

The ECU+ includes graphical tools for updating the firmware that runs inside of your ECU+ silver box. These tools are described below.

9.1 Upgrading the DSP Firmware

The ECU+ includes two microprocessors inside the silver box. The first microprocessor is the so-called “DSP” microprocessor. This microprocessor runs the engine and does the majority of the datalogging.

To upgrade the DSP firmware, execute the following steps:

1. Connect your laptop to your ECU+ silver box.
2. Turn on the ignition, but don't start the car.
3. Run the new ECU+ Win software by double-clicking its icon on your Windows desktop.
4. Use the *Settings->Update Firmware->DSP Firmware* menu item to start the graphical DSP firmware update wizard. This wizard will save your current head unit settings to disk, upgrade your firmware, and restore your settings. In most cases, this process is as simple as pressing Next a few times. Always upgrade to the latest DSP firmware (this is the wizard's default), as downgrading is not supported.

The firmware upgrade should complete without error. Some sample screenshots are shown below.

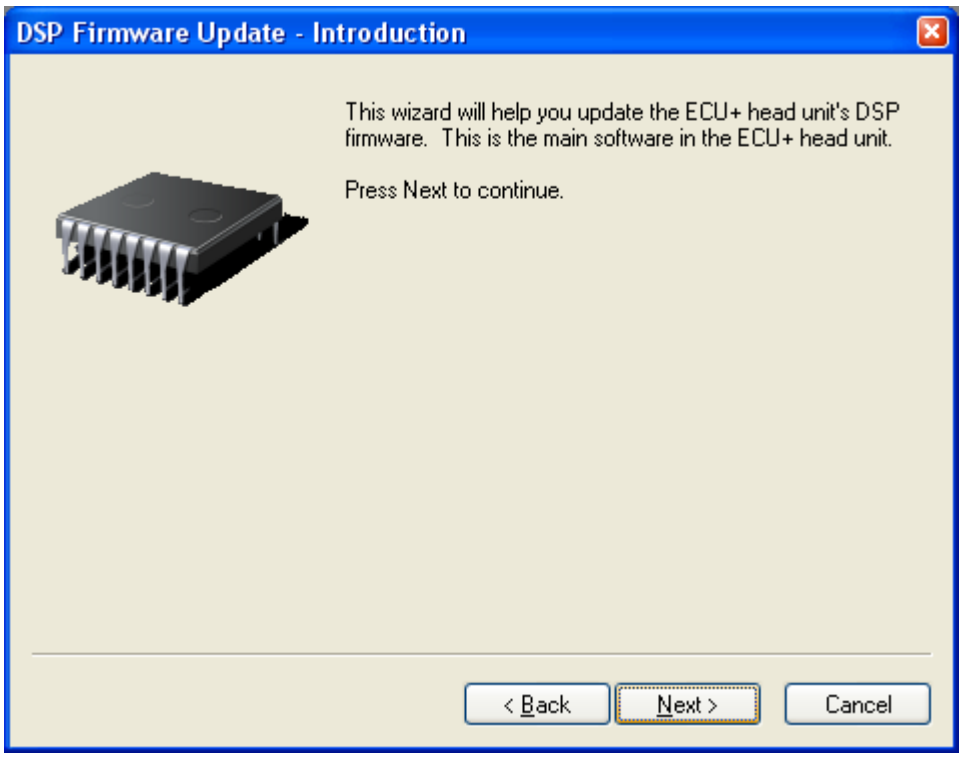


Illustration 117 - Starting the DSP firmware update process

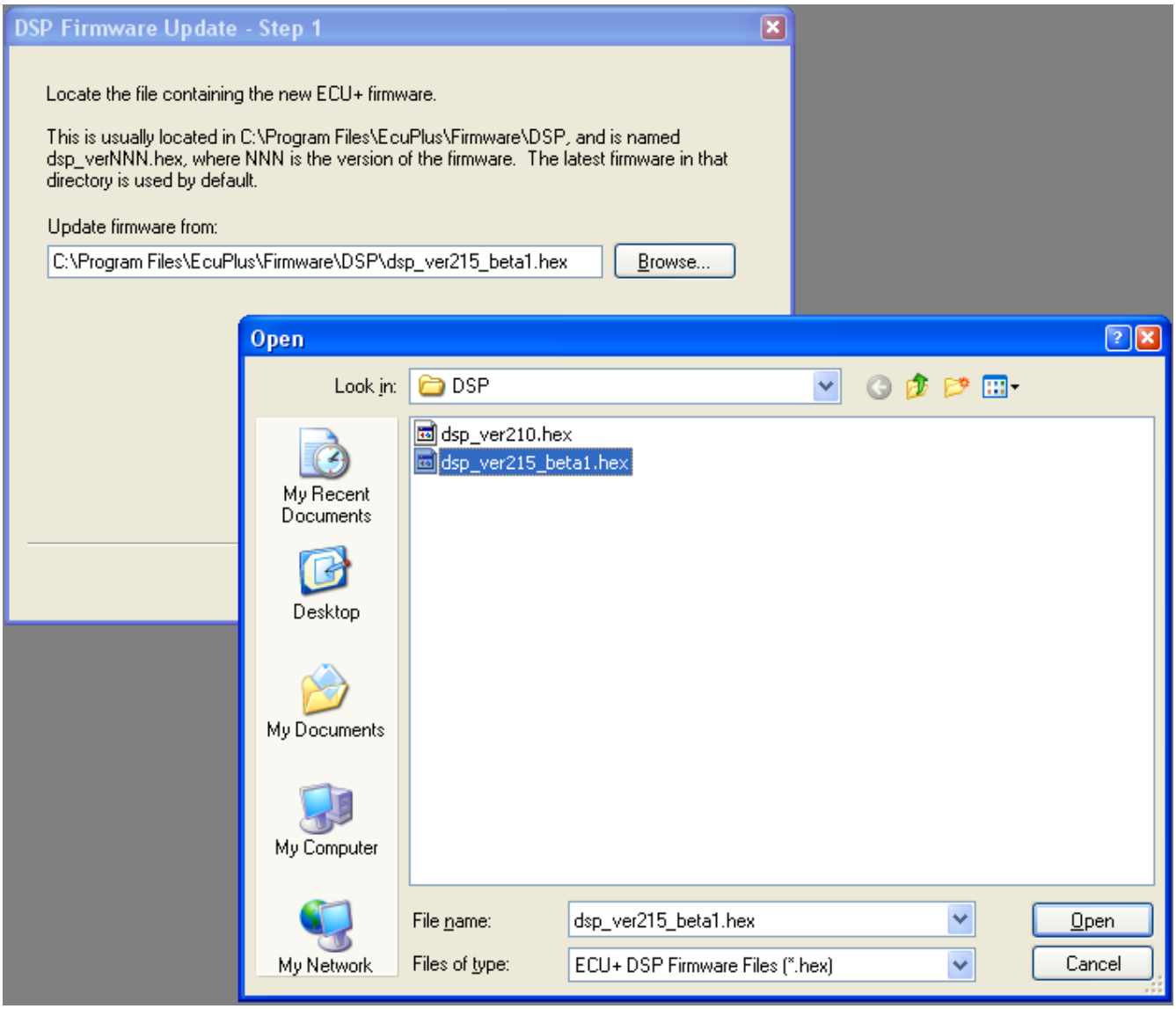


Illustration 118: Optionally selecting the firmware file

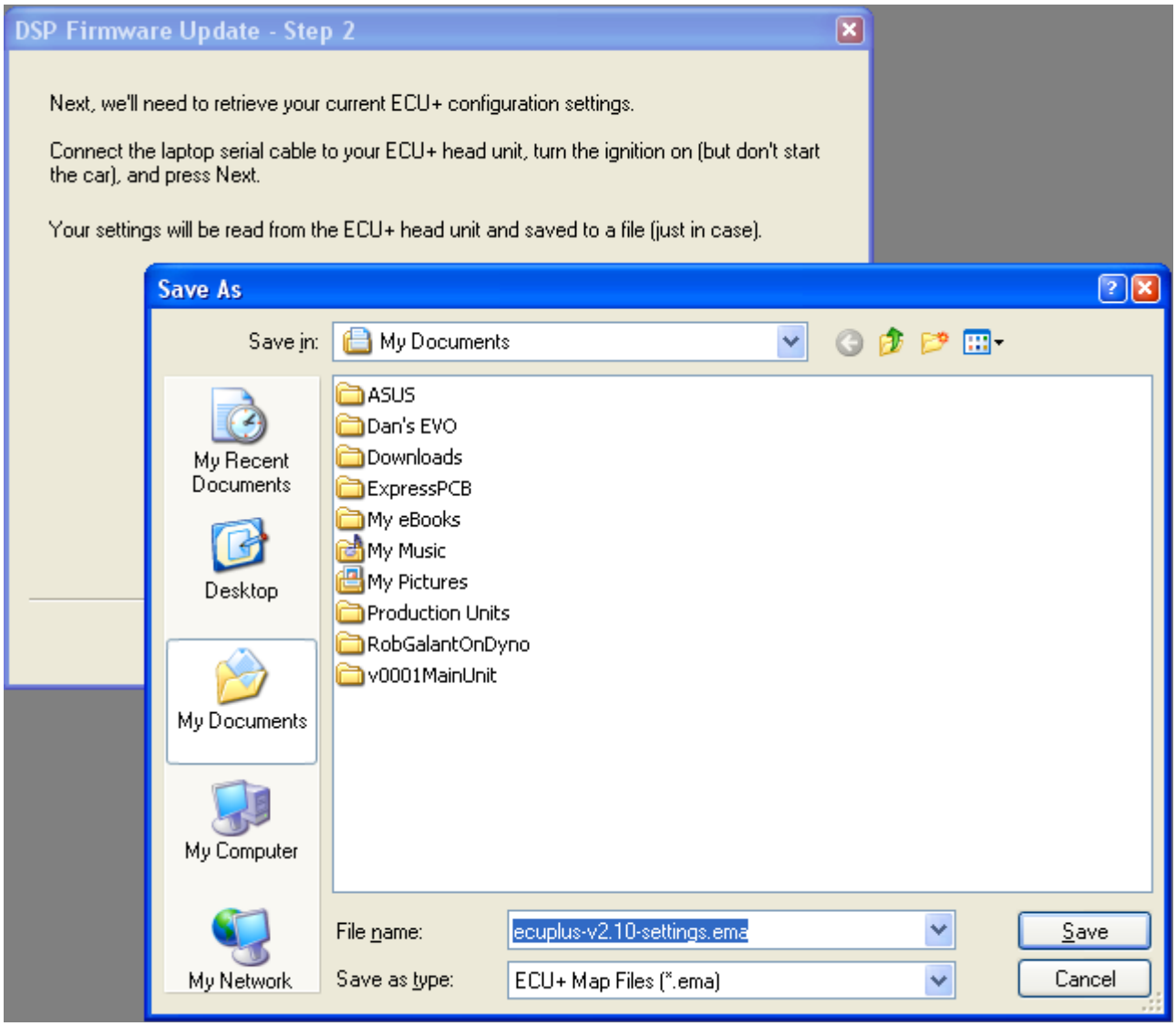


Illustration 119: Saving your current settings to a file

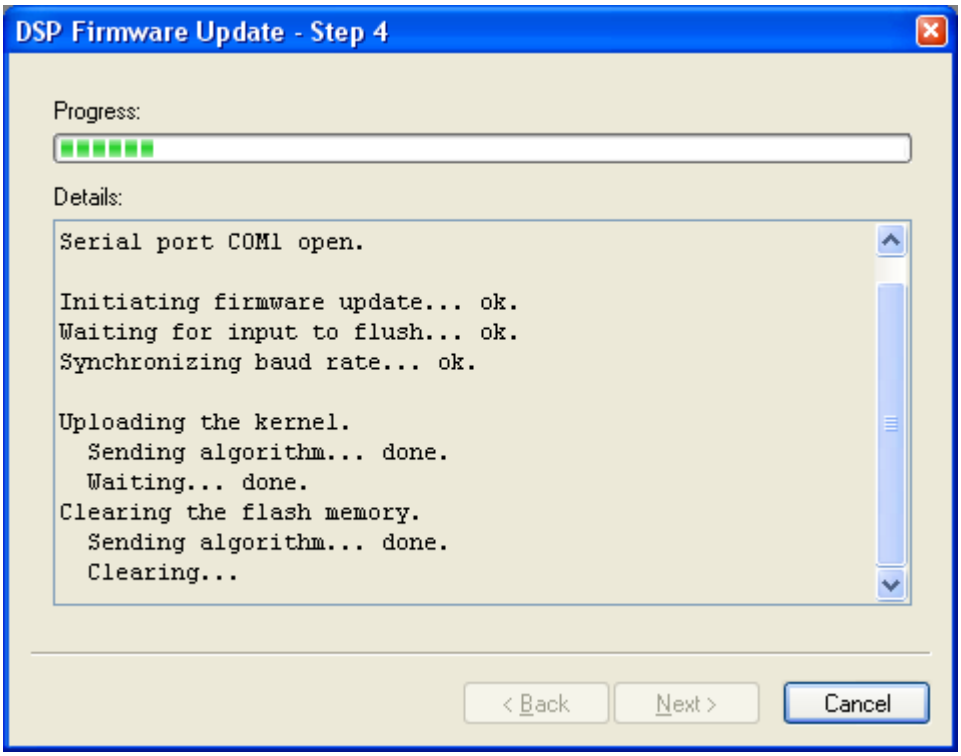


Illustration 120: Doing the upgrade

9.2 Upgrading the AVR Firmware

The second microprocessor inside the silver box is the so-called “AVR” microprocessor. This microprocessor talks to the OBD port on your stock ECU.

To upgrade the AVR firmware, execute the following steps:

1. Connect your laptop to your ECU+ silver box.
2. Turn on the ignition, but don't start the car.
3. Run the new ECU+ Win software by double-clicking its icon on your Windows desktop.
4. Use the *Settings->Update Firmware->AVR Firmware* menu item to start the graphical AVR firmware update dialog box.
5. Use the Browse button to select an AVR firmware version appropriate for your car. Separate versions are supplied for first generation DSMs, second generation DSMs (1995-1996), second generation DSMs (1997-1999) and the Evolutions.
6. Press the Start button to begin the upgrade.

The firmware upgrade should complete without error. A sample screen is shown below.

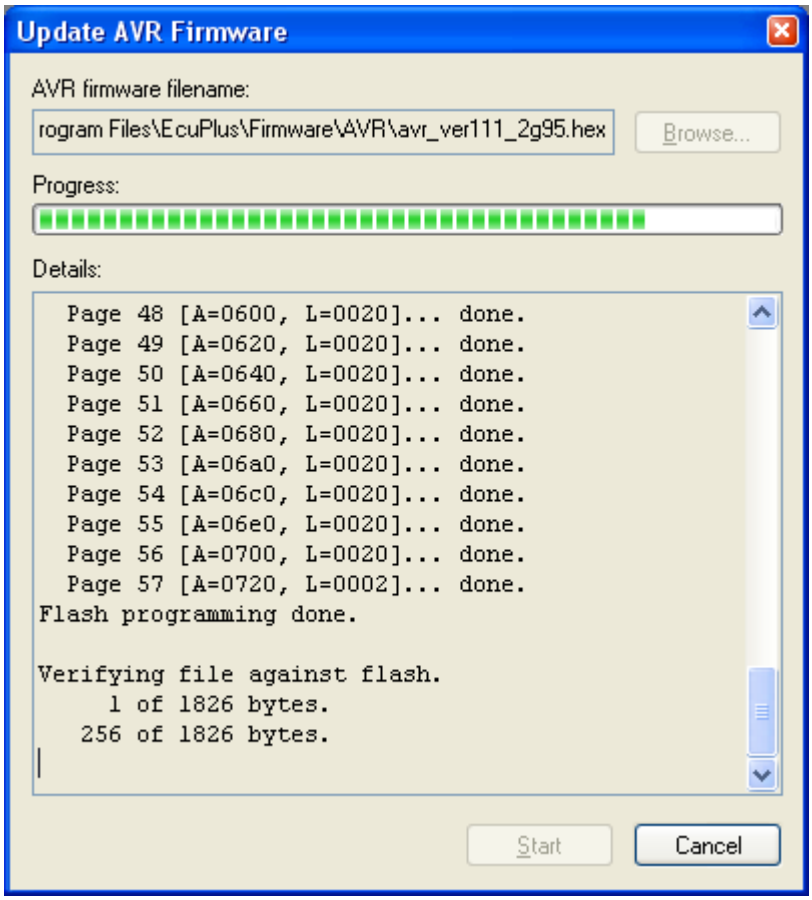


Illustration 121 - A sample run of the AVR firmware update

10 Using the ECU+ with a GM MAF Meter

The ECU+ has the ability to work directly with a 3” or 3.5” GM hotwire-style MAF meter in place of the stock MAS on the DSMs or Evolutions. This capability, which is similar to that provided in the popular MAF Translator device, replaces the stock MAS with the lower-restriction GM unit. The ECU+ will accept the GM MAF's output signal directly, and “translate” it into a signal compatible with the stock ECU.

You can use the GM MAF in place of the stock MAS in the intake (between the air filter and the turbo), or in a “blow through” configuration in which the stock MAS is removed and the GM MAF is placed in the upper intercooler pipe between the compressor bypass valve and the throttle body. The latter configuration has the side benefit of dealing well with compressor bypass valves that are vented to atmosphere (VTA).

Below are the steps required to use this GM MAF setup:

1. Purchase the GM MAF of choice. Both the 3” and 3.5” MAFs are available from Mach V Motorsports as well as other vendors for under $150.
2. Purchase a “pigtail” wiring adapter from Casper's Electronics, Inc (http://www.casperselectronics.com). This item, part #108098, is called a “LT1 MAF splice,” and it plugs into the GM MAF and breaks out the three electrical signals on the MAF.
3. Plumb the GM MAF into your intake, in either the normal or blow through mode. Make sure that the GM MAF is facing in the proper direction. The unit has an arrow on the body that indicates the

direction of air flow.

4. Make the electrical connections to the pigtail connected to the GM MAF, as follows:

Illustration 122- The LT1 MAF Splice "pigtail"

- Pigtail black wire to ground.
- Pigtail pink wire to a +12v source under-hood. Use a digital multimeter to verify the +12v signal before connecting it up.
- Pigtail yellow wire to the ECU+'s MAS input wire (this is pin 4, orange, Group A on the ECU+). To do this, extend the yellow wire through the firewall and into the passenger compartment. Then disconnect the ECU+'s pin 4 and connect it to the extended pigtail yellow wire. This is the GM MAF's output, which you're connecting directly to the ECU+.

5. When you remove the stock MAS, you're also removing the stock air temperature and barometric pressure sensors, which are built into the stock DSM or Evolution MAS unit. You need to make two more wiring changes that allow the ECU+ to output simulated air temperature and barometric pressure sensor signals:

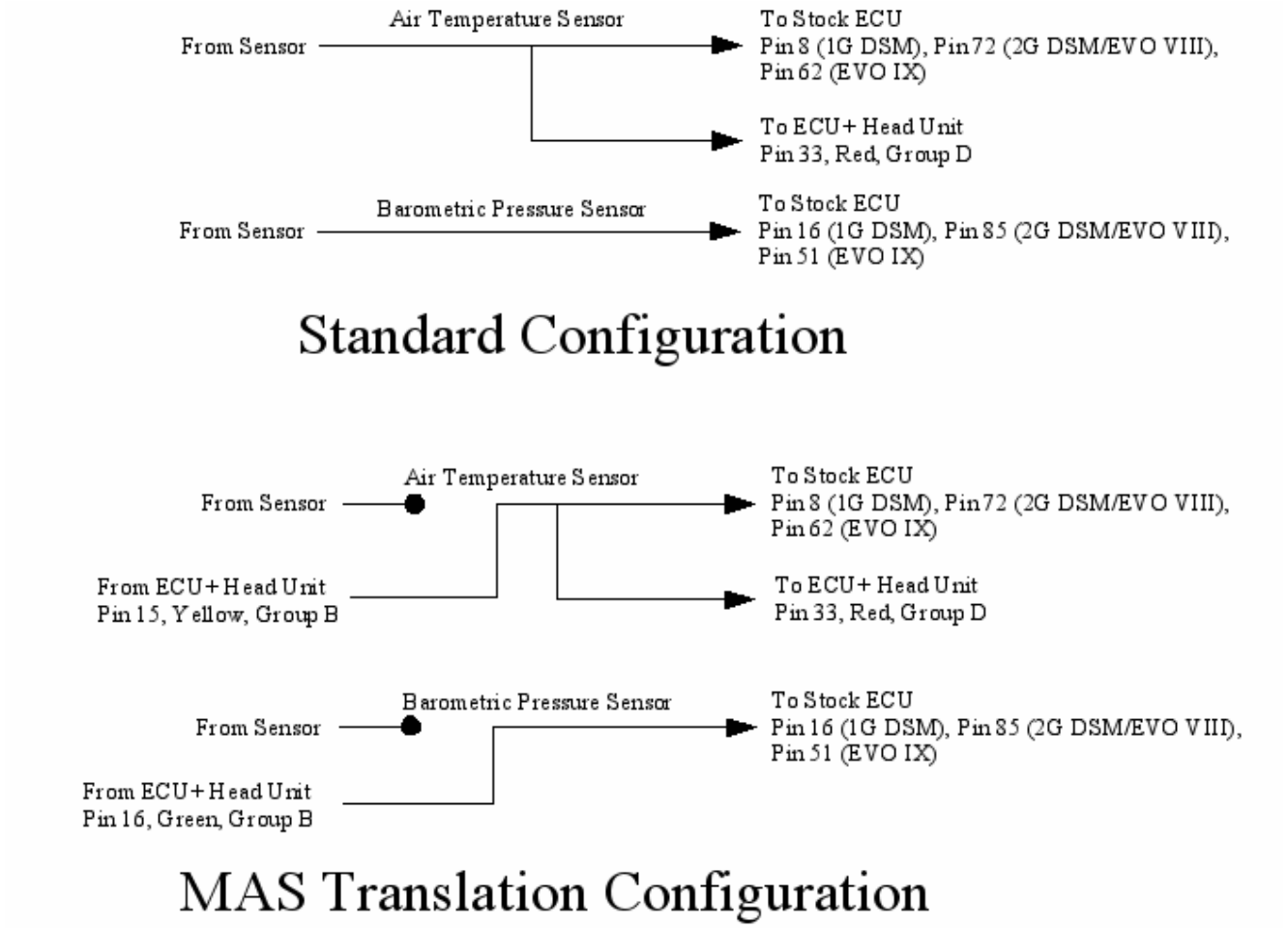


Illustration 123- The analog wiring changes for MAS translation

- Cut the air temperature wire (formerly) between the stock MAS and the stock ECU. This is pin 8 on the 1G DSM ECU, pin 72 on the 2G DSM and Evolution VIII ECUs, or pin 62 on the Evoution IX ECU. Solder the ECU+'s analog output 0 (pin 15, yellow wire, Group B) to the stock ECU air temperature input. If you've purchased a PnP ECU+, you can cut the pin 72 (2G DSMs and Evolution VIII) or pin 62 (Evolution IX) wire in the ECU+'s PnP harness.
- Optionally, connect the ECU+'s analog output 0 back into the ECU+'s air temperature datalogging input (pin 33, red wire, group D) as shown in the above illustration. If you cut the air temperature wire to the "left" (in the above illustration) of the tee in the air temperature wire, you get this connection for free. If you choose to not connect this signal back to the ECU+'s air temperature datalogging input, the ECU+ will display strange values for the air temperature, though this won't affect how your vehicle runs.
- Repeat the first step for the barometric pressure sensor wire. Cut it at the stock ECU (or in the PnP wiring harness) and connect this stock ECU input (pin 16 on the 1G DSM ECU, pin 85 on the 2G DSM and Evolution VIII ECUs, or pin 51 on the Evolution IX ECU) to the ECU+'s analog output 1 (pin 16, green wire, Group B).

6. Double-check your connections. When you're confident that the connections are ok, turn your ignition to the On position (don't start the car yet) and use the ECU+ Win software to navigate to the MAS Translation setup screen. Select the type of translation that you need, based on the type of GM MAF you've purchased, and your car type. Save this setting.

7. Finally, using the ECU+ Win software, navigate to the Analog Outputs screen, and set analog output 0 to 2.37 volts and analog output 1 to 3.90 volts. Save this setting.

Once all of the above are complete, start your car. The car should start and run normally, just like it did before the installation of the GM MAF. At this point, the ECU+ is doing the MAS translation continuously and fooling the stock ECU into thinking it still has a stock MAS connected. If you look at a capture, you should see MAS input and output frequencies that are very similar to that of the stock MAS for a given RPM and load combination.

With the GM MAF in place, you should drive your car for a while and verify that the car runs normally. Before doing any WOT runs, though, it's recommended that you (slowly) go through the steps of tuning your car again for optimal air-fuel ratio. The MAS translation in the ECU+ should only require minor changes to your fuel maps from the values used before installing the GM MAF, but it never hurts to re-verify your fuel map settings to account for any non-linearities of the stock MAS that have been corrected through the use of the GM MAF. In general, the GM MAF will flow more air before becoming non-linear and introduce much less restriction in the intake than the stock MAS.

Note: when using the GM MAF, you may notice that the input and output MAS frequencies never go to zero. This is because the GM MAF is especially sensitive, and outputs a low frequency signal even when the engine is not running.

11 Configuring and Using the ECU+'s Boost Controller

The ECU+ includes the ability to control boost using a GM boost control solenoid. Setting up boost control is one of the most complex tasks in the ECU+. This section supplies step-by-step instructions on how to set it up.

1. Install your boost control solenoid as described in the section Connecting a Boost Control Solenoid on page 27.
2. You must have a MAP sensor to control boost with the ECU+. Make sure that your MAP sensor is installed and working properly.
3. ECU+ boost control also assumes that the ECU+ can properly calculate what gear your vehicle is in, so make sure that the Gear Calculation settings are correct and working.
4. Locate a long, empty stretch of road where you can test your boost control settings.
5. Configure the ECU+ as follows:
    - Fuel and timing maps: very conservative. Rich on the fuel side with non-aggressive timing.
    - Map set switching: primary maps only.
    - Boost control settings:
        - *Wastegate frequency*: 30 Hz.
        - *Enable WG above TPS*: 0 volts. This causes the ECU+ wastegate maps to be used at all times, with no spool control.
        - *Overboost level*: 5 PSI above the boost level you want to run.
        - *Minimum WG duty cycle*: 5, *maximum WG duty cycle*: 95.
        - *Spool hysteresis*: 2 PSI.
    - Boost spool maps: set all cells to 0 PSI.
    - Boost wastegate maps: set all cells to 0% duty cycle.

6. Next, go out and make a very quick (perhaps ½ second) high-throttle run starting at 3000 or so RPM in 2nd gear. Verify that your boost level rises to a low pressure level – 10-14 PSI, and holds. This pressure level is set by your wastegate actuator spring, and varies depending on your turbocharger type. If you get 10-14 PSI in this test, then you're in good shape. If not, you may get a large boost spike. If this happens, your wastegate solenoid is either not electrically wired correctly, defective, or not receiving pressure from your pressure source.

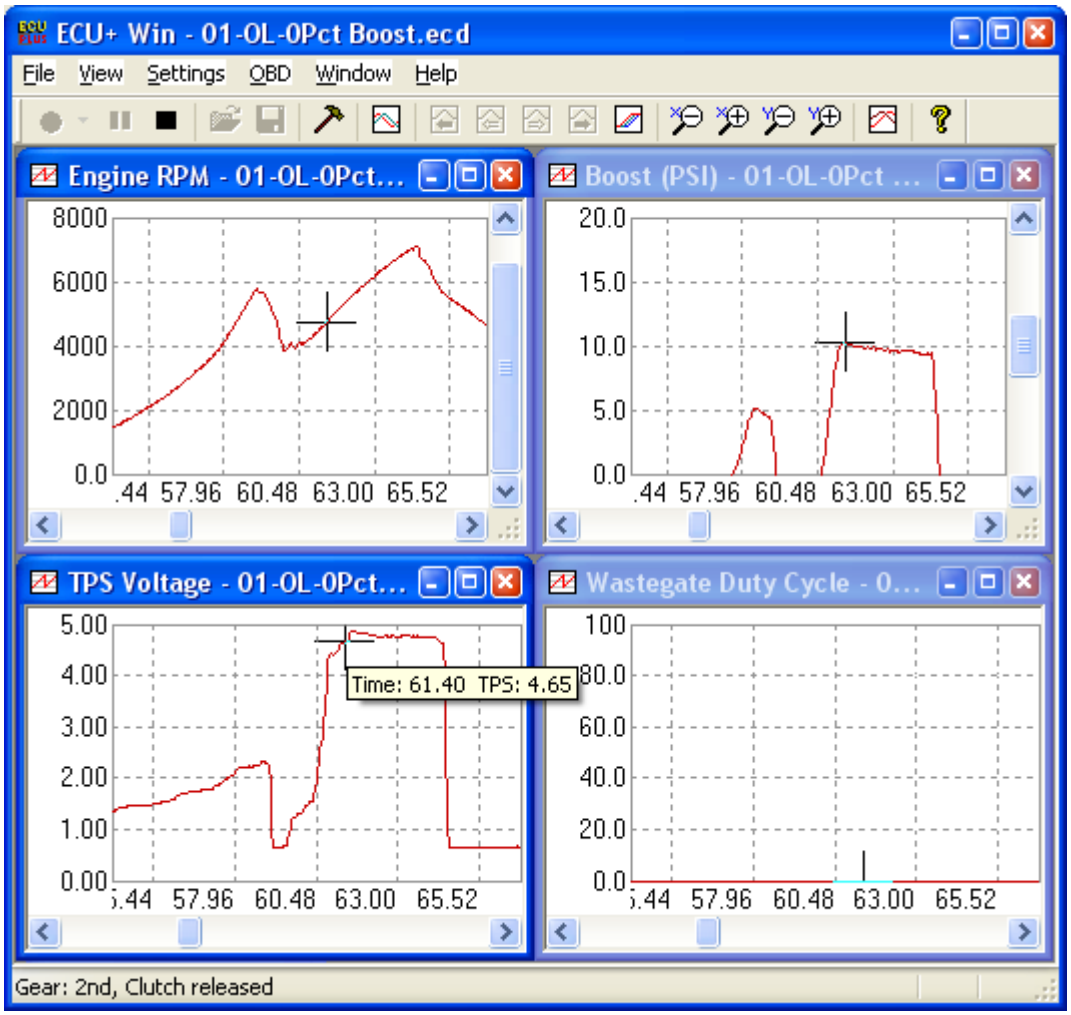


Illustration 124 - Boost with 0% wastegate duty cycle on an Evolution VIII

7. With this configuration up and working correctly, it's time to increase the wastegate duty cycle some. Increasing the wastegate duty cycle will increase boost, so make changes slowly and carefully. With all of the other settings as-is, increase all of the cells in the Boost wastegate maps to 20% and make another high-throttle, short 2nd gear pull while datalogging. This should make little to no difference in your boost level.
8. Again, increase all of the cells in the Boost wastegate maps to a larger value, say 40%. Make another pull and datalog that. Continue increasing the values and making more pulls. Eventually, your boost level will begin to increase with higher wastegate duty cycles. Below is the result of 60% duty cycle on a stock-turbo'd Evolution VIII in three gears.

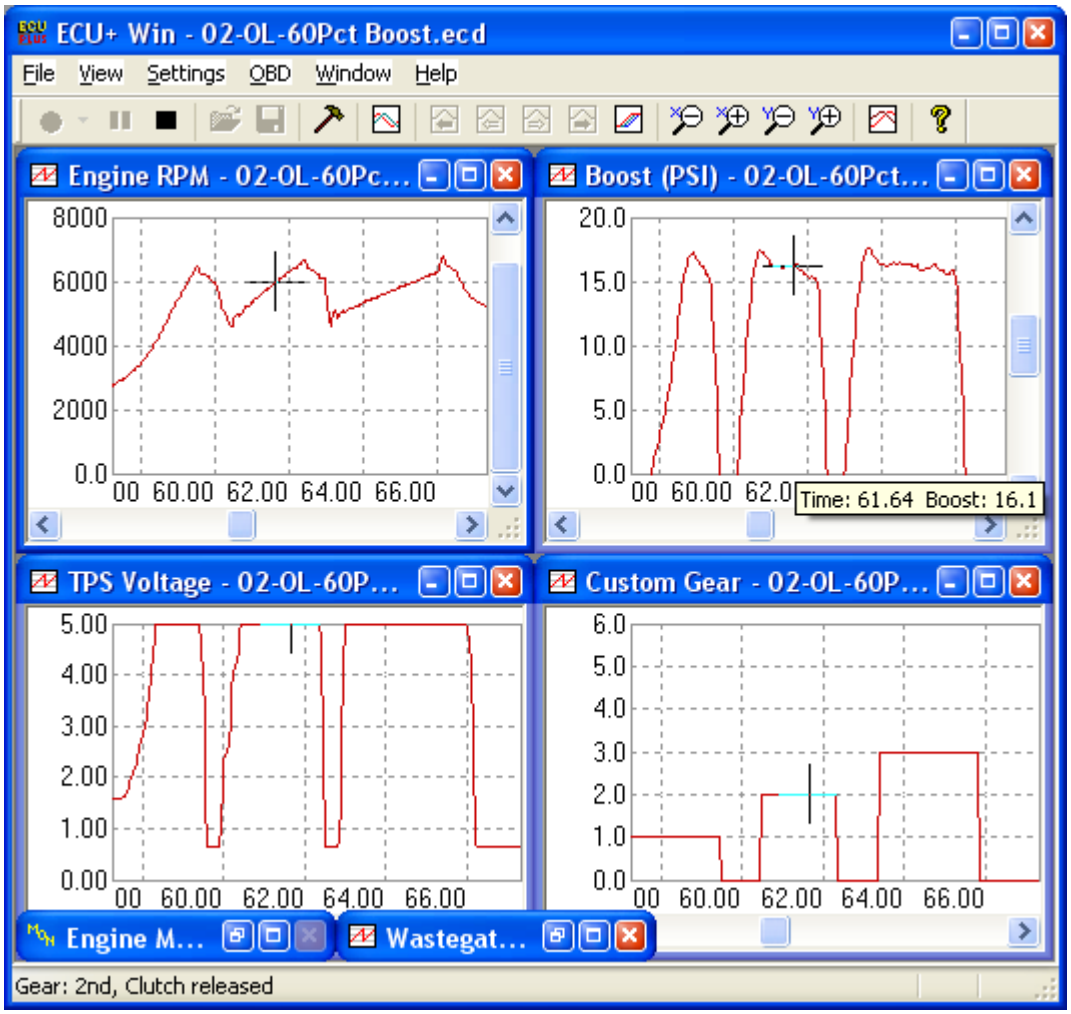


Illustration 125 - 60% duty cycle boost control, 1st-3rd

9. Eventually, you'll start to reach your target 2nd gear boost level. You may find that your boost levels fall off some at higher RPMs due to inefficiencies in your turbocharger. To correct this, make more pulls in 2nd, but very slightly adjust the Boost wastegate duty cycle cells higher at higher RPMs. Make at most a 2% change per run, and always start your pulls at the same RPMs at the exact same spot on the road. You may end up with settings like Illustration 126 below:

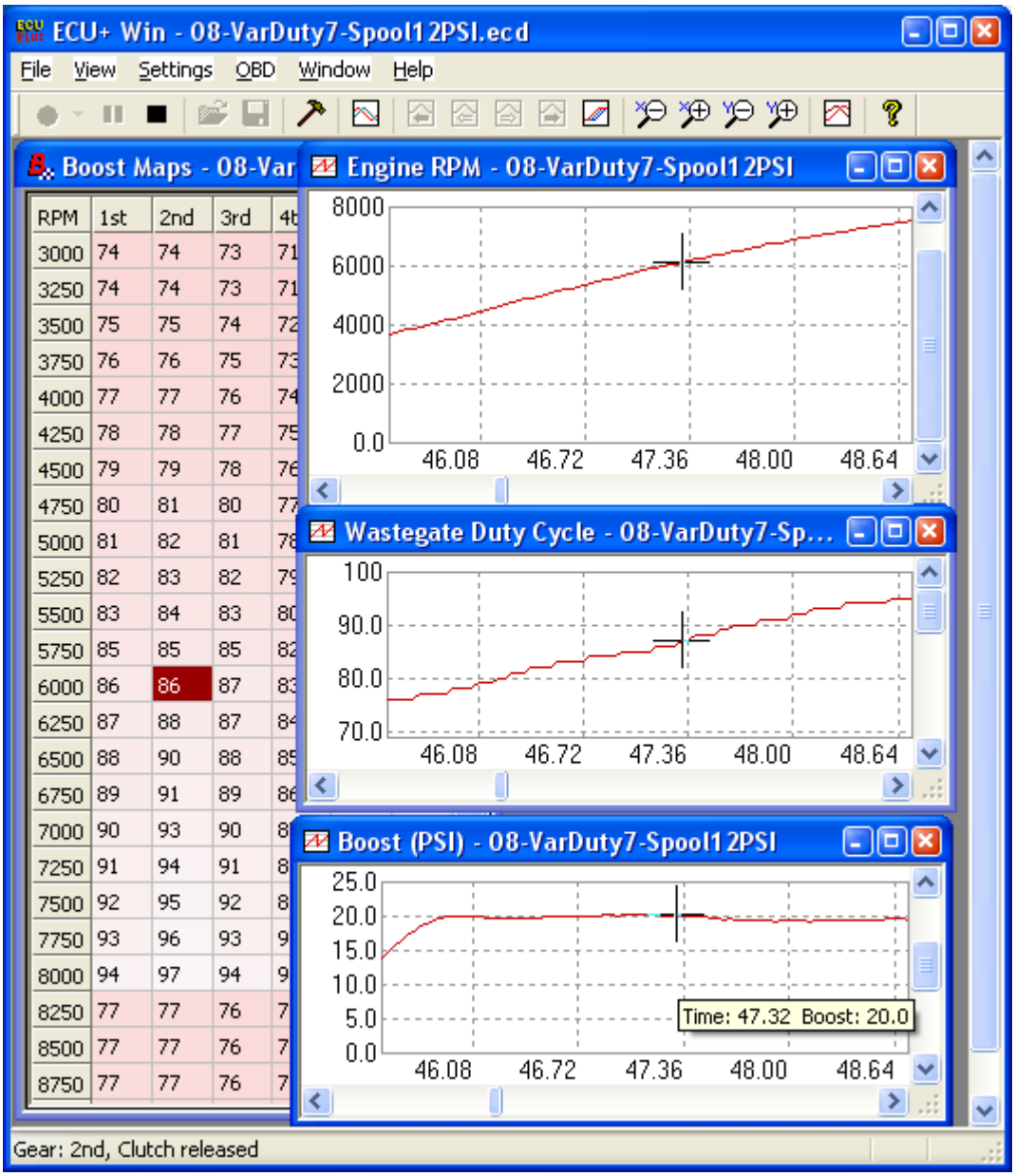


Illustration 126 - Near-final 20 PSI boost wastegate maps

Note how the wastegate duty cycle increases with RPM to maintain the target 20 PSI boost level.

10. With 2nd gear taken care of, copy the 2nd gear cells in the Boost wastegate maps to the 3rd gear column. Select the entire 2nd gear column by clicking on the 2nd gear column header, right-click, and select *Copy*. Then select the 3rd gear column, right-click and select *Paste*. Make some more pulls in 3rd and tweak the 3rd gear column as needed. Repeat this procedure for the other applicable gears.

11. At this point, bring up the Boost control settings dialog, and change the *Enable WG above TPS* setting back to 2 volts. The ECU+ uses this value as a minimum throttle position value. Below this TPS level, the ECU+ doesn't energize the boost control solenoid at all (driving it at 0% duty cycle for minimum (not zero) boost). Above this value, normal boost control operation is activated. Switching this back to 2 volts saves wear and tear on the boost control solenoid, and is highly recommended.

12. The ECU+ is now controlling your boost level, and it's easy to raise or lower your boost levels by manipulating the Boost wastegate maps table. The only downside to the way boost is being controlled now is that boost spool-up – the time when the turbocharger is spooling upward towards your target boost level – is not as fast as it would be with a manual boost controller. To fix that, you'll need to modify your Boost spool maps.

    The Boost spool maps are a second boost control table that allows the ECU+ to spool your turbo as quickly as possible while keeping boost spikes (overshoots of the target boost level) to a minimum. Note that it's not 100% possible to control boost spikes, and doing so limits how fast the turbo will spool. The ECU+'s boost spool maps allow you to make that trade-off between fast spool time and boost spikes.

    To begin setting up the boost spool maps, open that Head Unit Settings dialog and set all of the cells to a pressure value of 6 PSI below your target boost level in 2nd gear. Make a 2nd gear pull starting at 2500 RPM.

13. Bring up your 2nd gear pull in the ECU+ Win software. You should see something like this:

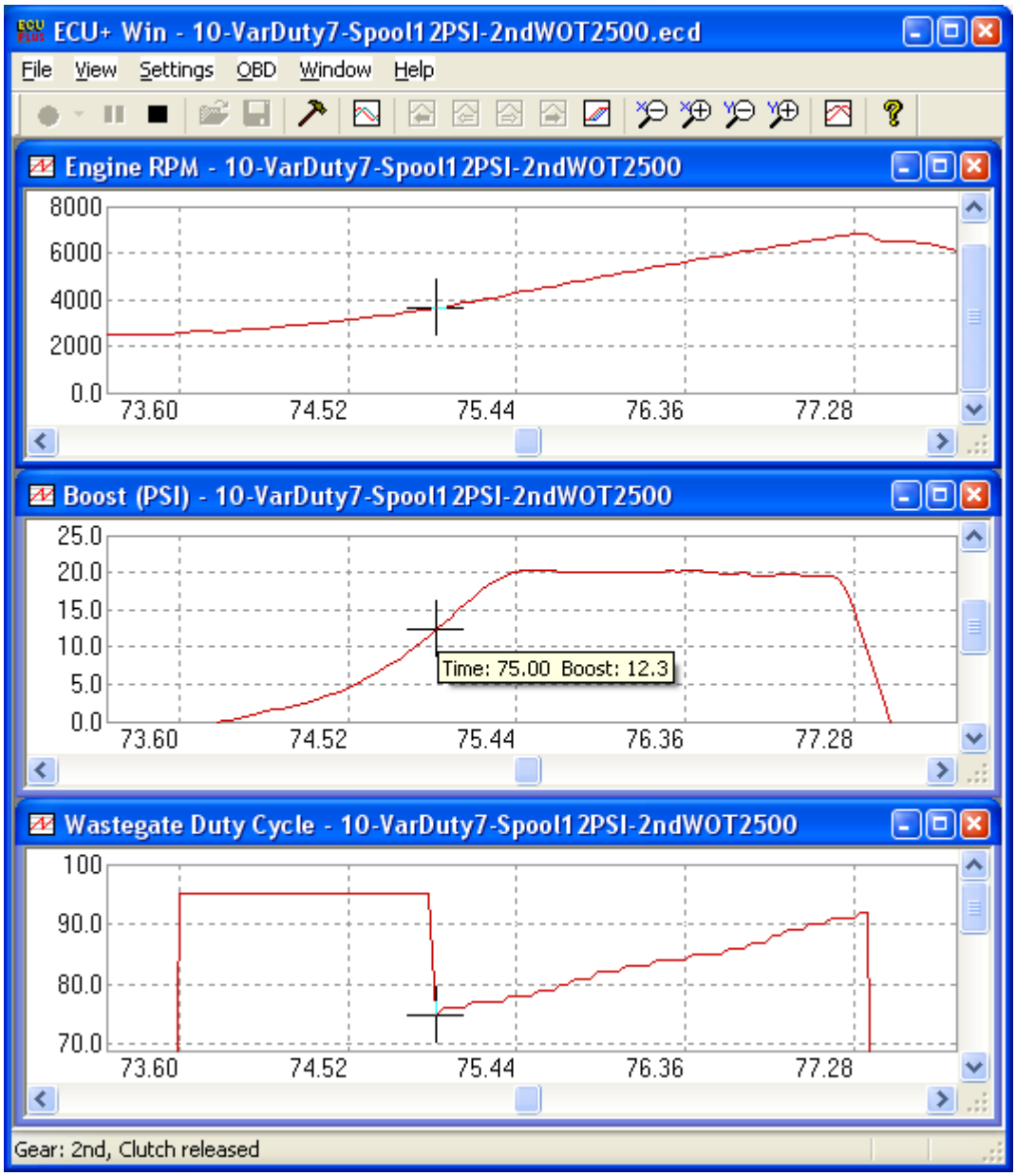


Illustration 127 - A 2nd gear pull with the spool map entries set to 12 PSI

In this pull, all of the Boost spool map cells were set to 12 PSI (8 PSI below the target of 20 PSI). You can see the wastegate duty cycle jumps to 100% when the throttle exceeds 2 volts. It then stays at 100% until the boost level exceeded the 12 PSI spool threshold, and then jumps down to 75% or so. What's happening is that the ECU+ is spooling the turbocharger below the spool threshold by driving the boost control solenoid at 100% - this makes the turbo spool as quickly as possible. As soon as the measured boost level exceeds the spool threshold, the ECU+ switches to the duty cycle in the Boost wastegate maps table, and boost holds steady from there on out.

14. Next, repeat the above 2nd gear pulls (again, use the exact same stretch of road and the exact same starting place on the road), but do pulls starting at 3000, 3500 and 4000 RPM. Bring up these captures in the ECU+ Win software. You may find that your no-spike runs now have a small spike, since the turbo will spool much quicker when starting at higher RPMs. Below is an example, showing the exact same settings as above, but starting at different RPMs. Note that the boost graph is significantly zoomed in to show the small boost spikes.

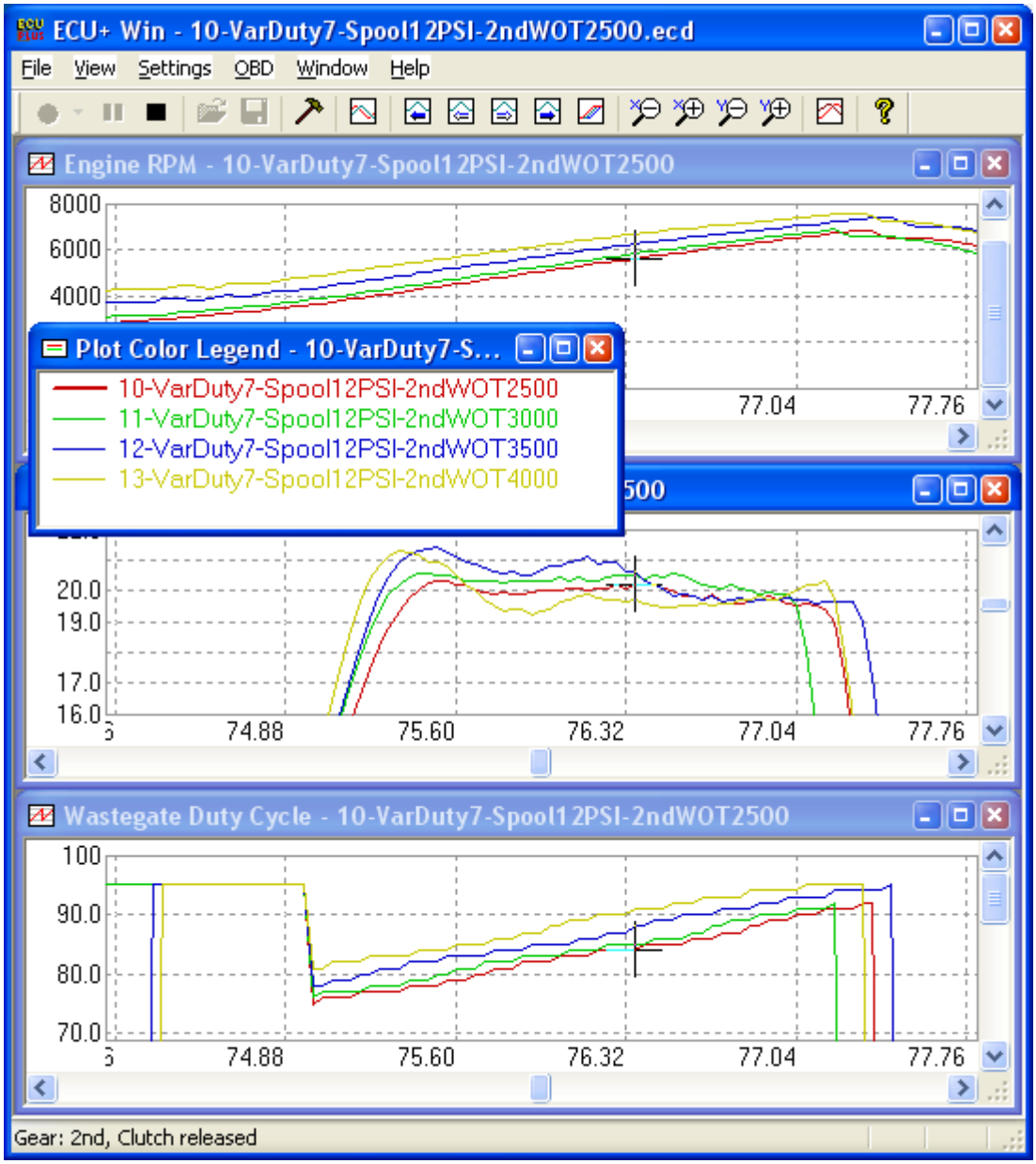


Illustration 128 - Small boost spikes when starting at 3000, 3500 and 4000 RPM

15. If the boost spikes are acceptable, leave the Boost spool maps as-is. If you'd like to reduce the boost spikes, alter the Boost spool maps as follows:
    - Open up the capture that starts at 4000 RPM in the ECU+ Win software.
    - Locate the RPM where the logged wastegate duty value drops from 100% down to a lower value. Make a note of this. For this example, let's assume that's 4800 RPM and that the value in the spool table at 4750 RPM (the closest one to 4800 RPM) is 12 PSI.
    - Open up the capture that starts at 2500 RPM in the ECU+ Win software and do the same thing. Let's assume the wastegate duty cycle drops down from 100% at 3500 RPM with 12 PSI in the spool table at that RPM, and that the 2500 RPM capture doesn't have a significant boost spike.
    - Add a "tilt" to your 2nd gear boost spool maps. Before making the change, all of the cells were at 12 PSI. The tilt reduces the spool boost level starting at 3500 RPM, and reduces the

values further as you go progressively higher in the table. Below is a sample.

RPM	1st	2nd	3rd	4th	5th
2000	12.0	12.0	12.0	12.0	12.0
2250	12.0	12.0	12.0	12.0	12.0
2500	12.0	12.0	12.0	12.0	12.0
2750	12.0	12.0	12.0	12.0	12.0
3000	12.0	12.0	12.0	12.0	12.0
3250	12.0	12.0	12.0	12.0	12.0
3500	12.0	12.0	12.0	12.0	12.0
3750	12.0	**11.2**	12.0	12.0	12.0
4000	12.0	**10.4**	12.0	12.0	12.0
4250	12.0	**9.6**	12.0	12.0	12.0
4500	12.0	**8.8**	12.0	12.0	12.0
4750	12.0	**8.0**	12.0	12.0	12.0
5000	12.0	**7.2**	12.0	12.0	12.0

Illustration 129 - Adding a tilt to the boost spool maps for 2nd gear

This change causes the Boost wastegate maps to kick-in sooner (at lower boost levels) when the boost is spooling starting at higher RPMs. This will reduce the boost spikes when spooling from a higher RPM. Add more tilt to control spikes more aggressively. Alternately, if you'd like even faster spool, increase the pressure levels in this table to allow faster spool with a higher spike. Be very careful when increasing the boost levels, though, and always keep the Boost wastegate maps at least 4-5 PSI below your target boost levels in each gear.

Once your ECU+ boost control is set up, it should provide very high quality control of your vehicle's boost level. After changing boost levels, always go back and re-tune your vehicle's air/fuel ratio and timing settings, as changing boost levels can have a severe impact on your tune.

12 Tuning Your Engine With the ECU+

This section provides some general information on tuning your car with the ECU+. Before starting, please read and heed the following warning:

YOU CAN DESTROY YOUR ENGINE WITH THE ECU+!

Neither the manufacturer nor the distributor of this device are responsible for any damage done to your vehicle due to the use of this device.

Always remember the above when tuning your vehicle. If you're conservative when tuning, you'll be able to use the ECU+ to make small, incremental improvements to your vehicle's performance, and most important of all, not destroy your engine in the process. It's sometimes said that engines make maximum horsepower just before they blow up...

To avoid destroying your engine, keep in mind that an excessively lean mixture and timing that is too-far advanced can lead to detonation. Excessive detonation will destroy your engine's internals. So be very very careful when reducing fuel flow or advancing timing with the ECU+ system.

12.1 “Tools” Required

When tuning your car, you should have the following “tools” available:

1. The ECU+ system (of course), with the ECU+ Win software running on an in-car laptop.
2. A boost gauge, and possibly the ECU+ map sensor, wired into the system.
3. An Exhaust Gas Temperature (EGT) gauge. This allows you to monitor your vehicle's exhaust gas temperature. Rapidly increasing EGT values often indicate detonation, which you can also monitor with the ECU+'s knock logging capability.
4. A wideband O2 sensor kit. These are moderately-priced, standalone boxes that can display your vehicle's air-fuel ratio very accurately. If this isn't available, the front O2 sensor in your vehicle will make a (poor) substitute.
5. A friend to ride in the car with you. Your friend will pilot the laptop while you pilot the car. Note: please don't refer to your friend as a “tool” - he might not appreciate it. :)

12.2 Making Runs

The best way to tune with the ECU+ is to make a series of dyno runs. See the section Dyno Analysis on page 72. You should select a flat, deserted road and make a series of runs, one per capture file, while giving the car a consistent few minutes to cool down between runs. Between each run, make a single change in the fuel, timing or other parameters. Then pull off the road and use the dyno analysis capabilities of the software to compare the most recent run to the previous run and see how the horsepower and torque curves compare. This will tell you whether you're making adjustments in the right direction or not.

You should start out by striving for consistency. The same section of road. Going WOT at exactly the same RPM and launch point on the road. The same dyno numbers and curves when you don't change anything. Once you've got a consistent set of dyno plots, change the fuel, or timing, and make some more runs. See if the car gained horsepower in a portion of the graph, or if it lost horsepower. Continue iterating until you've got the best performance from your engine.

12.3 General Tuning Techniques

- Before beginning tuning, set up the injector scaling values. These establish a baseline fuel offset for the injectors in your car.
- You should start with the injector scaling configured appropriately for your stock vs current injector sizes.
- With the injector scaling values in place, set all of the cells in the fuel and timing maps to all zero.
- Tune your boost levels first, then your fuel and timing maps.
- Tweak the idle fuel map cell (as well as the values in the MAS tweaks configuration screen) for best idle. You'll probably find that the larger the current injector size, the harder it is to achieve a smooth idle.
- On the road, make several runs with the fuel and timing maps set to zero. Try to launch your car from a roll consistently so that the resultant dyno plots are almost identical.
- Next, make changes to the fuel maps for WOT conditions and do some more runs. Power is generally gained by reducing the fuel map values, thus making the engine run leaner. From the factory, most cars run very rich at WOT, and the ECU+ allows you to compensate for that. For

each change you make, make no more than a 2% fuel map change from the last run. Change one or two fuel map cells at a time. When you bring the car to a stop, you'll want to look at the front O2 voltage at WOT, which should be in the range of 0.90-0.92 volts. Also look at the timing view and see how it compares to the previous run. If you find that the timing suddenly drops off at high RPMs, or that the knock display shows "spikes," there's a good chance that the stock ECU heard some knock from the motor. This indicates that you're running too lean – fix that right away. Also, when you're doing the runs, always watch your EGT gauge for rapidly rising temperature. Again, this indicates an overly-lean condition. If you have a wideband O2 sensor hooked up to the ECU+, use that for tuning. Try to get your air/fuel ratios (at WOT) in the 11:1 range on pump gas, and 12:1 on race fuel.

- Once you've optimized the WOT fuel tables, transfer the same values to the high load fuel table cells.
- With the fuel table optimized, try tweaking the timing tables at WOT. Make only 1-2 degree timing changes between runs. Generally, you'll want to advance the engine timing, thus moving the spark time backwards so that it fires earlier before top-dead-center (TDC) of the piston. This means putting positive numbers in the timing map cells. As with the fuel tables, make small changes and see what happens while watching for knock, high EGTs and timing retard.

13 Troubleshooting Software Problems

This section provides some tips for diagnosing and fixing common software-related problems.

13.1 ECU+ Win Problems

13.1.1 The Software Never Connects

When I click on the "New Capture" button, a "Connecting" box pops up that says "Connecting to the ECU+ head unit...", but a connection never happens.

To "connect," the ECU+ Win software looks for a specific stream of data from the head unit. Whenever the head unit is turned on, it transmits this stream of data. If the ECU+ Win software never sees the stream of data, it'll never "connect."

There are several things you'll want to check:

1. Is the ECU+ hooked up to the serial port? You'll use the straight-through DB-9 serial port to connect to your laptop. Double-check the connections, ensuring that the serial cable is secure and tightened down on both the ECU+ head unit and laptop end.
2. Are you connected to the serial port on your laptop. On a laptop, serial ports are male DB-9 connectors.
3. Does Windows recognize your serial port, and is it ok? Bring up the Windows Device Manager (usually under My Computer->Properties->Hardware->Device Manager). You should have a "Ports" entry for your serial port, and if you display its properties, Windows should say "this device is working properly."
4. Make sure the ECU+ Win software is using the correct serial port.
5. Make sure there are no hardware conflicts. Sometimes Windows will "detect" a serial mouse when none exists. Manually disable these in the device manager.

6. Your car's ignition must be in the on (not accessory) position to power up the ECU+ head unit.

As a last resort, you can use the SerTestW application to see if a ECU+ head unit seems to be connected to some serial port. This utility is covered in the section The Serial Port Tester, SerTestW on page 125.

13.1.2 Glitches

I sometimes get glitches on my capture files when using the ECU+ Win software. What's up?

The ECU+ Win application is a fairly CPU-intensive piece of software. If your laptop is slow, ECU+ Win may not be able to keep up with the incoming data stream. This causes “glitches” to appear in the captures. There are several things you can do to improve this:

1. Close all other applications when doing a capture.
2. Don't resize or move windows while doing a capture.
3. Within ECU+ Win, try to keep the number of graph views to a minimum. Redrawing the graphs takes the most CPU time of anything in the ECU+ Win application.
4. Minimize the fuel and timing maps, if visible. These also require significant CPU time to redraw in real time.
5. Minimize the ECU+ Win application while doing a capture.
6. Run a minimum of other software when using the ECU+ Win application. Common CPU hogs include applications that show up in your system tray (next to the clock, lower right hand corner of the screen) and anti-virus software.

13.1.3 Can't Open Serial Port

When I start a capture, ECU+ Win says it can't open the serial port. What could be wrong?

There are several possible reasons for this error. First, is the ECU+ Win software setup for the correct serial port? Step through the diagnosis in the section “The Software Never Connects” first. Once you've established that you have the correct serial port configuration, check to see if some other device is using the serial port. Often, the Palm desktop and HotSync applications will take over the serial port. Disable these applications and see if the problem goes away.

13.1.4 Boost Readings Don't Match My Boost Gauge

The ECU+ logs don't match the readings that I see on my boost gauge. I'm using the GM (or AEM) MAP sensor. Who's right?

They're both right. They're just measuring different things.

Both the GM and AEM sensors are MAP sensors. MAP stands for manifold **absolute** pressure – it measures absolute air pressure. Your boost gauge measures differential pressure – pressure as compared to ambient pressure. When your boost gauge reads "0 PSI," that means that the intake is at the same pressure as ambient. With the GM or AEM MAP sensors, "0 PSI boost" on the ECU+ means that you have 14.7 PSI of actual air pressure in your intake. This has two implications:

- If you're not at sea level, a MAP sensor will read out wrong based on your elevation (height above sea level). In Dulles VA where Mach V Motorsports is, the MAP sensor will be off by about 0.2 PSI. In Denver CO, it'll be off by 2.5 PSI. Always. If you drive from Dulles to Denver, it'll vary during the trip. If you look up your city at http://www.wunderground.com and get your elevation, you can see the MAP vs boost gauge inherent pressure error by plugging your elevation into the form at http://www.uigi.com/Atmos_pressure.html.
- The weather will affect your reading. Variations in weather will give you numbers that can be off by around +/- 0.5 PSI. You know when they say on TV that there's a high pressure system moving in? When it moves in, your MAP sensor reading moves up.

In addition to the above, all sensors have a accuracy that varies depending on pressure and temperature. The GM 3 bar unit, for example, will vary by +/-0.75 PSI, and will also vary on a per-trial basis by about 0.5 PSI for a new unit.

The bottom line is that your MAP sensor may or may not match what your boost gauge reads depending upon elevation and the accuracy of the sensor. It's possible to adjust your MAP sensor calibration so that the two will match, but it will take a bit of work. Simply configure your MAP sensor as a custom type, and adjust the calibration until the two readouts match.

14 Support and Getting Help

Because the ECU+ system is a complex device, you may have problems with the software or installation. You have several lines of support open to you:

- For hardware installation problems, contact the dealer that you purchased the ECU+ system from. They're best equipped to handle these type of problems.
- For software support, again your dealer is a good first place to start.
- For tuning help, your best bet is the forums at forums.ecuplus.com. The forums are a great place to hang out and learn about how others are using their ECU+. The forums are also used to announce new software or firmware updates, and you can request new features or ask almost any type of question here. If you own an ECU+, you should **definitely** monitor the forums for the latest ECU+ news and information.
- For any type of support, you also have the option of e-mailing the ECU+ developer – just send an e-mail to support@ecuplus.com. E-mails will generally be answered within 24 hours.
- If your ECU+ was working fine, but “breaks,” contact your dealer for information about their return and repair policies.